

PROPOSTA DE ALGORITMO DE OTIMIZAÇÃO DA INCLINAÇÃO DE FUROS EM DESMONTE A CÉU ABERTO

FRANCISCO EDUARDO ALMEIDA DA SILVA

Dissertação submetida para satisfação parcial dos requisitos do grau de
MESTRE EM ENGENHARIA DE MINAS E GEO-AMBIENTE

Orientador: Professor Doutor Alexandre Júlio Machado Leite

Coorientador: Mestre Vinícius Gouveia de Miranda

OUTUBRO DE 2018

VERSÃO DE DISCUSSÃO

MESTRADO INTEGRADO EM ENGENHARIA DE MINAS E GEO-AMBIENTE 2017/2018

DEPARTAMENTO DE ENGENHARIA DE MINAS

Tel. +351-22-508 1986

+351-22-508 1960

Fax +351-22-508 1448

✉ minas@fe.up.pt

Editado por

FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

Rua Dr. Roberto Frias

4200-465 PORTO

Portugal

Tel. +351-22-508 1400

Fax +351-22-508 1440

✉ feup@fe.up.pt

🌐 <http://www.fe.up.pt>

Reproduções parciais deste documento serão autorizadas na condição que seja mencionado o Autor e feita referência a *Mestrado Integrado em Engenharia de Minas e Geo-Ambiente - 2017/2018 - Departamento de Engenharia de Minas, Faculdade de Engenharia da Universidade do Porto, Porto, Portugal, 2018*.

As opiniões e informações incluídas neste documento representam unicamente o ponto de vista do respetivo Autor, não podendo o Editor aceitar qualquer responsabilidade legal ou outra em relação a erros ou omissões que possam existir.

Este documento foi produzido a partir de versão eletrónica fornecida pelo respetivo Autor.

À minha Mãe, aos meus Padrinhos, e ao Bentinho

L'esperienza ha poco da insegnare se non viene vissuta con umiltà.

Michelangelo Buonarroti

AGRADECIMENTOS

Findo o meu percurso na Faculdade de Engenharia com a entrega deste trabalho, não poderia de deixar de agradecer a algumas pessoas que foram fundamentais para a realização do mesmo, mas também fundamentais durante todo o meu percurso académico:

- Ao meu Orientador, o Professor Alexandre Leite, não só pelo seu contributo na realização desta Dissertação, mas também por todo o apoio prestado ao longo do curso, e por toda a pedagogia e esforço em melhorar a aprendizagem no curso de Minas;
- Ao meu Coorientador, Eng. Vinícius Miranda, meu Professor e mentor, por toda a sabedoria transmitida, e pelo tempo dispensado na elaboração deste trabalho, um Obrigado não é suficiente;
- A todos os docentes do Departamento de Minas que contribuíram para a minha aprendizagem ao longo dos últimos quatro anos;
- A toda a equipa da O-Pitblast, Lda., pelo bom ambiente com que sempre fui recebido, e por todo o apoio prestado, em especial à Eng. Raquel Sobral por toda a disponibilidade e paciência dispensada nas deslocações ao terreno;
- À equipa da dst group, em especial ao Diogo Fonseca, pela amabilidade em ceder o espaço para que possamos ter tentado realizar a parte experimental;
- Aos meus amigos de Civil, pelos primeiros quatro anos da minha vida académica, e em especial aos “Gormitis” e à família Rangers do Texas... que continuemos a fazer história!
- Aos meus amigos de Minas, em especial os “Discentes”. Que possamos contar sempre uns com os outros na nossa vida profissional, como contámos ao longo destes últimos quatro anos;
- À Fabiana, pelos últimos cinco anos, pelo carinho, paciência, e apoio incondicional, fundamental para o meu sucesso;
- Aos meus Padrinhos, por tudo aquilo que fizeram por mim;
- À Alice e ao Bentinho, por todo o carinho, e por serem os avós que nunca tive oportunidade de conhecer;
- À minha Mãe, por tudo aquilo que me proporcionou, e por todos os impossíveis que fez para que chegasse até aqui.

RESUMO

A indústria mineira tem sido beneficiada nos últimos anos com a introdução de novas tecnologias, como o caso de equipamentos laser e drones, que permitem auxiliar os operadores no planeamento e execução de desmontes com recurso a explosivos. Como este tipo de desmonte acarreta, por vezes, algumas dificuldades a nível do controlo da energia utilizada, podem surgir problemas com a fragmentação do produto final, problemas de projeções e vibrações que podem afetar as estruturas circundantes ao local do desmonte.

Como tal, surge cada vez mais a necessidade de dar uso a estas novas tecnologias, nas quais se inclui o uso de inteligência artificial, permitindo poupar recursos, tempo e dinheiro e melhorando o desempenho do desmonte. Assim, surge também o tema desta Dissertação, cujo objetivo é a construção de um algoritmo capaz de obter uma inclinação ótima para furos de desmonte a céu aberto. Este algoritmo foi construído em MATLAB e Excel, com recurso a fundamentos de desmonte de rocha, programação linear, álgebra linear e geometria analítica, e permite dispensar todo o trabalho manual de dimensionamento de furos, sendo apenas necessário obter os dados geométricos da bancada. Após a sua execução, são obtidos automaticamente os dados relativos à inclinação e afastamento da primeira linha de furos.

A validação deste algoritmo em ambiente de campo, deverá permitir a melhoria da fragmentação e das projeções de material rochoso, sendo possível obter desmontes de rocha mais seguros e eficazes, com diminuição de custos em fragmentação secundária.

PALAVRAS-CHAVE: algoritmo, inclinação de furos, otimização, programação linear, inteligência artificial, desmonte de rocha a céu aberto

ABSTRACT

The mining industry has benefited with the introduction of new technologies in recent years, such as laser and drone equipment, which assist operators in the planning and execution of rock blasting procedures. Because, sometimes, rock blasting entails some difficulties in controlling the energy released by the explosives, problems may arise with inadequate fragmentation on the final product, and problems such as flyrock and vibrations that can affect the structures surrounding the site of the dismantling.

As such, there is an increasing need to use these new technologies, which include the use of artificial intelligence, saving resources, time and money and improving the performance of the blasting. Thus, the theme of this Dissertation arises, whose objective is the construction of an algorithm capable of obtaining an optimum inclination for open-pit boreholes. This algorithm was built in MATLAB and Excel, using various fundamentals which include rock blasting, linear programming, linear algebra and analytical geometry, and it allows to dispense all the work of manually dimensioning the borehole inclination, being only necessary to obtain the geometric data of the bench. After execution, the data concerning the inclination and burden of the first row of holes are automatically obtained.

The validation of this algorithm in a field environment should allow the improvement of the fragmentation results and the projection of flyrocks, being possible to obtain safer and more efficient rock blastings, with lower costs in secondary fragmentation.

KEYWORDS: algorithm, borehole inclination, optimization, linear programming, artificial intelligence, open-pit rock blasting

ÍNDICE GERAL

AGRADECIMENTOS.....	i
RESUMO.....	iii
ABSTRACT.....	v
ÍNDICE DE FIGURAS.....	x
1. INTRODUÇÃO.....	1
1.1. ENQUADRAMENTO E OBJETIVO DO TRABALHO.....	1
1.2. ORGANIZAÇÃO DO TRABALHO.....	2
2. DESMONTE DE ROCHA.....	3
2.1. INTRODUÇÃO E RESENHA HISTÓRICA.....	3
2.2. CARACTERIZAÇÃO DE MACIÇOS ROCHOSOS.....	4
2.3. EXPLOSIVO.....	6
2.3.1. EXPLOSIVOS INDUSTRIAIS.....	7
2.3.2. CONSUMO ESPECÍFICO (POWDER FACTOR).....	7
2.3.3. MECANISMOS DE ROTURA E DISTRIBUIÇÃO DE ENERGIA.....	8
2.4. VARIÁVEIS GEOMÉTRICAS CONTROLÁVEIS DE UMA PEGA DE FOGO.....	10
2.4.1. DIÂMETRO DO FURO - D.....	11
2.4.2. ALTURA DA BANCADA – H.....	12
2.4.3. AFASTAMENTO – B.....	12
2.4.4. INCLINAÇÃO DOS FUROS.....	13
2.4.5. TAMPONAMENTO – T.....	14
2.4.6. SUBFURAÇÃO – J.....	14
2.4.7. ESPAÇAMENTO – S.....	15
2.5. ACESSÓRIOS DE INICIAÇÃO E DETONAÇÃO.....	15
2.5.1. DETONADORES ELÉTRICOS.....	16
2.5.2. DETONADORES NÃO-ELÉTRICOS.....	17
2.5.3. DETONADORES ELETRÓNICOS.....	19
3. FUNDAMENTOS DE PROGRAMAÇÃO LINEAR.....	21
3.1. INVESTIGAÇÃO OPERACIONAL.....	21
3.1.1. RESENHA HISTÓRICA.....	21
3.1.2. FASES E TÉCNICAS DA INVESTIGAÇÃO OPERACIONAL.....	22
3.1.3. MODELAÇÃO MATEMÁTICA.....	22
3.1.3.1. Variáveis de decisão, restrições e função objetivo.....	23
3.1.3.2. Tipos de modelo matemático.....	23
3.2. PROGRAMAÇÃO LINEAR.....	23
3.2.1. SOLUÇÃO ÓTIMA.....	24
3.2.2. MÉTODO SIMPLEX.....	25

3.2.2.1. Solver do Microsoft Excel	25
3.2.2.2. Função <i>linprog</i> do MATLAB.....	25

4. MODELAÇÃO..... 27

4.1. DEFINIÇÃO DO PROBLEMA..... 27

4.2. TRATAMENTO DE DADOS EM MATLAB..... 30

4.2.1. REPRESENTAÇÃO GEOMÉTRICA DO TERRENO E REDUÇÃO DA AMOSTRA..... 30

4.2.2. REPRESENTAÇÃO DOS FUROS E DISCRETIZAÇÃO DOS PONTOS DE CADA FURO..... 32

4.2.3. OBTENÇÃO DO PERFIL CRÍTICO..... 35

4.2.4. DETECÇÃO E EXCLUSÃO DE *OUTLIERS*..... 37

4.2.5.1. Projeção vetorial de um ponto numa reta – função *projectPointinLine.m*..... 38

4.2.5.2. Detecção e exclusão dos *outliers* com recurso à função *EraseOutliers.m*..... 40

4.2.5. CÁLCULO DA DIREÇÃO DE DISPERSÃO DO PERFIL CRÍTICO – OBTENÇÃO DE NOVA BASE..... 41

4.2.5.1. Matriz de covariância..... 41

4.2.5.2. Valores e vetores próprios..... 42

4.2.6. MUDANÇA DE BASE..... 45

4.2.6.1. Base..... 45

4.2.6.2. Nova base..... 46

4.2.6.3. Conceito de mudança de base segundo Boulos (1987):..... 47

4.2.6.3. Mudança de base recorrendo à função *getPointsNewBase.m*:..... 48

4.2.7. TRANSLAÇÃO E PERFIL CRÍTICO FINAL..... 49

4.3. RESOLUÇÃO DO PROBLEMA UTILIZANDO PROGRAMAÇÃO LINEAR EM MATLAB..... 50

4.3.1. MODELO MATEMÁTICO DE UM FURO..... 50

4.3.2. CONSTRUÇÃO DO MODELO DO FURO..... 50

4.3.3. VARIÁVEIS DE DECISÃO..... 52

4.3.4. FUNÇÃO OBJETIVO..... 52

4.3.5. RESTRIÇÕES..... 53

4.3.6. RESOLUÇÃO COM A FUNÇÃO LINPROG..... 53

4.3.6.1. Otimização a partir do tampão..... 53

4.3.6.4. Resultado final da otimização..... 60

4.4. MODELAÇÃO E RESOLUÇÃO DO PROBLEMA UTILIZANDO PROGRAMAÇÃO LINEAR NO EXCEL..... 62

4.4.1. CONSTRUÇÃO DO MODELO DO FURO EM EXCEL..... 62

4.4.1.1. Preparação prévia de amostra de pontos..... 62

4.4.1.2. Modelo do furo e afastamento crítico..... 63

4.4.2. VARIÁVEIS DE DECISÃO, FUNÇÃO OBJETIVO E RESTRIÇÕES..... 64

4.4.3. SOLUÇÃO ÓTIMA APRESENTADA PELO ALGORITMO..... 65

5. SUGESTÕES PARA TRABALHOS FUTUROS COM ELABORAÇÃO DE CASO DE ESTUDO	69
5.1. NOTA PRÉVIA	69
5.2. DESCRIÇÃO DO PROCEDIMENTO EXPERIMENTAL A DESENVOLVER	69
5.2.1. SELEÇÃO DA BANCADA A DESMONTAR	69
5.2.2. LEVANTAMENTO TOPOGRÁFICO DA FRENTE LIVRE	70
5.2.1.1. Levantamento topográfico com recurso a perfilómetro laser	70
5.2.1.2. Obtenção de dados para execução do algoritmo	71
5.2.1.3. Levantamento topográfico com recurso a drone equipamento com câmara de alta resolução	72
5.2.3. EXECUÇÃO DO ALGORITMO	73
5.2.4. PERFURAÇÃO E VERIFICAÇÃO COM INSTRUMENTO MEDIDOR DE DESVIO DE FUROS	73
5.2.5. ANÁLISE DO DESMONTE	75
6. CONCLUSÕES	77
6.1. CONSIDERAÇÕES FINAIS	77
6.2. ALGUMAS SUGESTÕES PARA TRABALHOS FUTUROS	78
REFERÊNCIAS BIBLIOGRÁFICAS	79

ÍNDICE DE FIGURAS

Figura 1 - Pontos ótimos das relações “fragmentação/custo” e “fragmentação/impacte ambiental” (Bhandari, 1997)	4
Figura 2 – Gráfico de escavabilidade de maciços rochosos (Kaya, et al., 2011)	5
Figura 3 – Explosivos encartuchados (à esquerda) e a granel (à direita).....	6
Figura 4 - Mecanismos de rotura de maciços rochosos através do uso de explosivos (Jimeno, et al., 2003).....	8
Figura 5 – Gráfico Pressão vs Volume (Jimeno, et al., 2003).....	9
Figura 6 - Corte esquemático de bancada e furo	10
Figura 7 – Corte esquemático de um furo carregado e do perfil da frente livre (Hustrulid, 1999)	12
Figura 8 - <i>Booster</i>	15
Figura 9 – Corte esquemático de um detonador elétrico instantâneo (à esquerda) e com retardo (à direita)	16
Figura 10 – Aspeto exterior de um detonador elétrico	16
Figura 11 – Detonador não-elétrico (à esquerda) e conector (à direita) (Dyno Nobel, 2011).....	17
Figura 12 – Pegas de fogo com saída do material para o centro da bancada (Union Española de Explosivos, 1997)	18
Figura 13 – Pegas de fogo com saída do material para as laterais da bancada (Union Española de Explosivos, 1997)	18
Figura 14 – Unidade central (computador) de um sistema eletrónico de detonação	19
Figura 15 – Comparação entre sistemas elétricos e não-elétricos, com o sistema eletrónico (Cardu, et al., 2013).....	19
Figura 16 – Problema com múltiplas soluções ótimas (Hillier, 2010)	24
Figura 17 – Problema sem solução ótima (Hillier, 2010)	24
Figura 18 – Bancada modelada na plataforma O-Pitblast®.....	27
Figura 19 – Modelo de furo vertical gerado na plataforma O-Pitblast®.....	28
Figura 20 – Modelo de furo ajustado à frente livre	29
Figura 21 - Obtenção de dados de terreno e de furos	30
Figura 22 – Inserção de parâmetros de afastamento crítico e de tamponamento.....	31
Figura 23 – Exemplo de código para obtenção de uma amostra de pontos reduzida.....	31
Figura 24 – Comparação entre nuvens de pontos original (à esquerda) e reduzida (à direita)	31
Figura 25 – Representação esquemática do processo de criação de um furo com pontos discretizados	32
Figura 26 – Excerto de código para obter a representação dos furos discretizados	33
Figura 27 – Excerto de código relativo à adição de pontos extra	34
Figura 28 – Representação tridimensional da linha de furos no terreno	34
Figura 29 – Excerto de código para obtenção do perfil crítico	35
Figura 30 – Esquema representativo de um perfil de bancada com nuvem de pontos candidatos	36
Figura 31 – Pormenor da zona de influência do ponto P do furo, e distância entre um ponto candidato C	36
Figura 32 – Gráfico de uma reta de regressão correspondente a um conjunto de dados com um <i>outlier</i>	37
Figura 33 – Projeção de um ponto do perfil crítico na reta do furo	38
Figura 34 – Função <i>projectPointinLine.m</i>	39
Figura 35 – Função <i>eraseOutliers.m</i>	40
Figura 36 – Dispersão de um conjunto de pontos e vetores próprios associados.....	42
Figura 37 – Vetores próprios direcionais para o caso tridimensional	42
Figura 38 – Vetor próprio que indica a direção de dispersão de um conjunto de dados (representação bidimensional).....	43
Figura 39 – Cálculo da matriz de covariância e do vetor próprio associado ao maior valor próprio	44
Figura 40 – Vetor próprio que indica a direção de dispersão do perfil crítico (representação tridimensional)	44
Figura 41 – Representação gráfica de uma base canónica em \mathbb{R}^3	45
Figura 42 – Formação de base nova.....	46
Figura 43 – Representação gráfica dos novos eixos coordenados	46

Figura 44 – Função <code>getPointNewBase.m</code>	48
Figura 45 – Perfil crítico na nova base e rotação de 90° (representação bidimensional)	49
Figura 46 - Translação do perfil crítico (representação bidimensional)	49
Figura 47 – Afastamento entre pontos de frente livre (Y_n) e pontos discretizados de um furo vertical (X_n).....	51
Figura 48 - Representação esquemática do furo com a zona de tampão assinalada a vermelho.....	54
Figura 49 – Etapa inicial do processo de otimização, considerando apenas os pontos do furo abaixo da zona de tampão.....	54
Figura 50 – Excerto de código relativo à formulação da matriz A e vetor b	58
Figura 51 – Formulação do vetor f , relativo à função objetivo	59
Figura 52 – Excerto de algoritmo relativo à programação linear com função <i>linprog</i> e variáveis de saída.....	60
Figura 53 – Exemplo de <i>output</i> da matriz <i>result</i> para 11 furos	60
Figura 54 - Perfil crítico nº7 (representação 3D, à esquerda) e perfil e furo nº7 (representação 2D, à direita).....	61
Figura 55 – Perfil crítico nº11 (representação 3D, à esquerda) e perfil e furo nº11 (representação 2D, à direita).....	61
Figura 56 – Exemplo de redução da amostra de pontos no Microsoft Excel	62
Figura 57 – Tabela-padrão para inserção de dados	63
Figura 58 – Inserção de parâmetros no <i>Solver</i>	65
Figura 59 – Coordenadas da frente livre, definição da reta do furo, parâmetros de afastamento, função objetivo minimizada e variáveis de decisão otimizadas pelo <i>Solver</i>	66
Figura 60 – Representação gráfica em Excel, com o furo otimizado	66
Figura 61 – Quadro de parâmetros	67
Fig. 62 – Levantamento topográfico com perfilómetro laser	70
Figura 63 – Exportação do modelo para ficheiro CSV	71
Figura 64 – Levantamento topográfico com drone	72
Figura 65 - Modelo de bancada gerado com recurso a drone	72
Figura 66 - Boretrak® de barras	74
Figura 67 – Exemplo de furo real desviado relativamente ao teórico, com utilização de Boretrak	74
Figura 68 – Análise de fragmentação com Wipfrag	75

1

INTRODUÇÃO

1.1. ENQUADRAMENTO E OBJETIVO DO TRABALHO

No desmonte de rocha com recurso a explosivos, são conhecidos alguns dos impactes ambientais mais relevantes, nomeadamente, a geração de poeiras, vibrações e onda aérea e a projeção de blocos, cuja mitigação deve ser considerada como prioritária no que toca ao desenho da pega de fogo. Simultaneamente, é desejável que a pega de fogo seja o mais eficiente possível na obtenção de material com granulometria adequada às necessidades da produção, e que esta eficiência se traduza através do menor custo possível com o desmonte.

A indústria mineira, apesar de estar bastante desenvolvida no que diz respeito às técnicas de extração, lida ainda com bastantes problemas que podem ser evitados com o recurso a novas tecnologias e ferramentas informáticas que permitam otimizar alguns dos processos extrativos. Atualmente, outras indústrias fazem uso de ferramentas de inteligência artificial que permitem otimizar muitos dos seus processos, nunca dispensando um utilizador humano (seja engenheiro, ou qualquer outro tipo de operador) que supervisione o processo e faça a tomada de decisão final, de modo a garantir que o sistema esteja a operar de maneira correta, e a cumprir os objetivos propostos. Assim, surge este tema de dissertação, com o intuito de construir uma ferramenta que seja capaz de otimizar substancialmente a eficiência do desmonte de rocha com recurso a explosivos.

Através do uso de um algoritmo capaz de calcular a inclinação ótima dos furos de desmonte, devidamente ajustados à frente livre, deverá ser possível tornar o desenho de uma pega de fogo bastante mais eficiente não só no que toca ao tempo despendido a dimensionar os furos da pega, como também poderá trazer melhorias a nível da granulometria do produto obtido, e a nível dos impactes causados pelo desmonte como as projeções de material e a geração de blocos. Este trabalho contempla assim a construção desse mesmo modelo de otimização, mas também a validação do mesmo em ambiente de campo, de modo a verificar quais os efeitos causados pelo uso do mesmo, e se coincidem com o que é previsto e pretendido.

1.2. ORGANIZAÇÃO DO TRABALHO

Esta dissertação encontra-se dividida em seis capítulos, apresentando-se em seguida uma breve descrição de cada um desses capítulos:

- O presente capítulo é introdutório, com a finalidade de expor a premissa que leva à necessidade da elaboração deste trabalho;
- No capítulo 2 é feita uma breve revisão do estado da arte do desmonte de rocha, com os conceitos fundamentais a ter em conta mais adiante, tanto na construção do algoritmo e formulação do modelo de otimização, como durante a parte experimental da validação do mesmo;
- No capítulo 3 descrevem-se alguns dos fundamentos de programação linear essenciais à correta formulação do modelo de otimização adotado neste trabalho;
- No capítulo 4 é apresentado em detalhe o algoritmo de otimização de maneira sequencial, sendo suportado com excertos de código MATLAB e do ambiente em Excel de modo a facilitar a compreensão do mesmo. Ao longo do capítulo é feita uma descrição explícita dos conceitos de programação, álgebra linear, geometria analítica, desmonte de rocha, e programação linear, utilizados na construção do algoritmo, tanto em linguagem MATLAB como em Excel;
- O capítulo 5 debruça-se sobre as etapas necessárias para realizar a validação do modelo, recorrendo a instrumentos e técnicas correntes. A validação do modelo surge como sugestão para trabalhos futuros que permitam analisar a eficácia do modelo, através da análise de dados recolhidos em atividade experimental numa exploração a céu aberto,
- Por fim, no capítulo 6 são sintetizados os principais resultados obtidos e apresentadas as conclusões deste trabalho.
- Nos Anexos desta Dissertação, poderá ser consultado o código integral do algoritmo, e são expostos os resultados de dois casos práticos de modo a confirmar que o algoritmo se encontra a funcionar corretamente.

2

DESMONTE DE ROCHA

2.1. INTRODUÇÃO E RESENHA HISTÓRICA

A exploração de pedra pelo Homem remonta à Pré-História, com o aparecimento dos primeiros utensílios em pedra, essenciais para a sobrevivência dos primeiros seres humanos. Com a descoberta do fogo, surge também a fundição e o fabrico de utensílios metálicos, a partir de metais extraídos de determinadas rochas. Já na Antiguidade Clássica, as grandes civilizações dedicaram-se não só ao fabrico de utensílios e armas constituídas por cobre, ferro e bronze, mas também à extração de grandes volumes de rocha para construção de obras monumentais que no século XXI ainda se mantêm de pé, como são exemplo o Coliseu de Roma, ou as Pirâmides de Gizé.

A revolução na indústria extrativa deu-se com a aplicação da pólvora para extração mineira no séc. XVII e com a invenção da dinamite, por Alfred Nobel no séc. XIX. A utilização de explosivos permitiu a otimização do processo de desmonte de maciços mais resistentes, graças à energia libertada durante o processo de detonação dessas misturas químicas.

Atualmente, o desmonte de rocha pode ser efetuado com recurso a explosivos ou somente recorrendo a equipamentos que promovem a desagregação, de forma mecânica, dos maciços. O desmonte de rocha com recurso a explosivos, abordado nesta Dissertação, é usado não só na indústria extrativa, mas também em obras geotécnicas como é o caso da construção de túneis, portos, barragens hidroelétricas, vias rodoviárias e ferroviárias, e ainda em obras civis de demolição de edifícios (Sanchidrián & Muñiz, 2000). No caso da indústria extrativa, o principal objetivo é a obtenção de um recurso natural existente na crosta terrestre. Poderá tratar-se de um minério presente num determinado jazigo mineral, ou então uma rocha extraída na sua globalidade e transformada numa granulometria adequado para determinados fins. Há assim, três grandes tipos de explorações mineiras: minas a céu aberto, minas subterrâneas e pedreiras.

No caso das minas, e principalmente no caso das minas metálicas, o minério encontra-se presente e disseminado em grandes massas rochosas heterogêneas sem valor económico (ganga), sendo necessário proceder à sua extração dando especial atenção ao fator de diluição do material, de modo a que a exploração seja economicamente viável (Domingo, et al., 2015). No caso das pedreiras o minério é a própria massa mineral do jazigo e, no caso das pedreiras destinadas à produção de agregados, é necessário ter em especial atenção o grau de fragmentação do material obtido.

O desmonte de rocha com explosivos é desenhado e projetado com recurso a diagramas de fogo, cujos princípios gerais de dimensionamento são abordados neste capítulo. A elaboração de um diagrama de fogo consiste no desenho de uma geometria de furação e cálculo de uma determinada quantidade e respetiva distribuição de explosivo a usar, tendo como objetivo o desmonte de um volume de rocha que,

no caso desta Dissertação, corresponderá a um certo volume de rocha em bancada a céu aberto. As variáveis de dimensionamento de um diagrama de fogo podem ser não controláveis (caso das características geológicas e geotécnicas do maciço rochoso), ou controláveis. Jimeno (2003), define três tipos de variáveis controláveis num diagrama de fogo:

- Explosivo;
- Geometria do diagrama de fogo (furação);
- Tempos de retardo entre detonações de furos consecutivos e sequência de iniciação.

As variáveis passíveis de serem controladas num desmonte de rocha devem ser calculadas de modo a que o dimensionamento da pega de fogo seja ótimo, isto é, deverá ser atingido um custo mínimo nas operações de perfuração, rebentamento, carga e transporte do material desmontado e tratamento/transformação do minério em simultâneo com a obtenção de um grau de fragmentação adequado (McKenzie, 1966). Para além disso, há que conjugar todas as variáveis de projeto com as variáveis económicas, de modo a garantir que o desmonte tenha um custo mínimo e minimizando simultaneamente alguns efeitos indesejados como sejam as projeções de rocha, a onda aérea, o ruído, as vibrações e geração de poeiras (Bhandari, 1997).

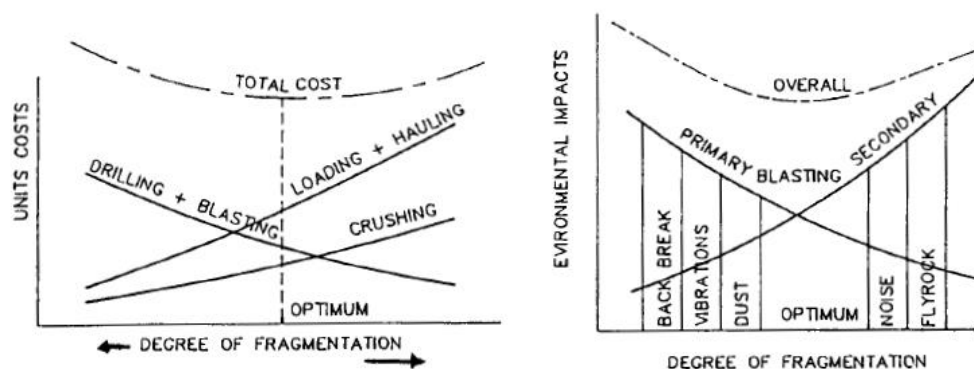


Figura 1 - Pontos ótimos das relações “fragmentação/custo” e “fragmentação/impacte ambiental” (Bhandari, 1997)

2.2. CARACTERIZAÇÃO DE MACIÇOS ROCHOSOS

A primeira fase do dimensionamento de uma pega de fogo é a caracterização o mais completa possível da área onde vai decorrer o desmonte. Para tal é necessário fazer o levantamento topográfico da bancada em que vai ocorrer a atividade de desmonte, bem como realizar estudos de prospeção geológica e geofísica tendo em vista a determinação das propriedades do maciço, em particular da densidade e geometria da fraturação do mesmo, fatores muitas vezes condicionantes do modo de atuação dos explosivos no momento de detonação.

O levantamento topográfico deverá ser feito com um perfilómetro laser, drone, ou outro equipamento semelhante, e tem como objetivo definir e modelar a topografia da bancada, passando-se a conhecer algumas das suas características geométricas. Deste modo, poder-se-á obter dados para a construção de modelos digitais do terreno. Atualmente, estes dados podem ser importados e utilizados em alguns softwares que, para além de construírem os referidos modelos digitais de terreno, sobre eles podem fazer

o dimensionamento do diagrama de fogo e prever o produto obtido consoante alguns dos parâmetros recolhidos e escolhidos para o desmonte (caso do O-Pitblast®).

Os estudos de prospeção geológica e geofísica são implementados para melhoria do conhecimento da litologia local, para a determinação da densidade e características da fraturação, em particular sobre a existência de falhas, zonas mineralizadas e a presença de água no maciço. É também necessário prever o comportamento mecânico do material rochoso quando submetido a tensões de rotura, através da utilização de classificações geomecânicas baseadas na colheita de parâmetros obtidos por observação direta, ensaios *in situ* ou amostras recolhidas por sondagens de prospeção. Uma das classificações geomecânicas mais utilizadas é o sistema *Rock Mass Rating* (RMR) de Bieniawski (1989), em que são atribuídos pesos a alguns parâmetros inerentes ao maciço rochoso, tais como: resistência à compressão uniaxial, RQD, influência da água e o espaçamento, condição e orientação das descontinuidades. Após a atribuição dos pesos de cada parâmetro referido, é feito o seu somatório e é obtido um índice RMR, que corresponderá a uma classe de maciço rochoso. Estas classificações são de amplo uso prático na caracterização de maciços rochosos, tendo especial aplicabilidade também no caso dos túneis e explorações subterrâneas, onde são de extrema importância como informação para que se garanta a estabilidade das cavidades geradas por desmonte.

Outros métodos utilizados englobam a caracterização da escavabilidade do maciço, como o método de Franklin et al., que relaciona a resistência à compressão uniaxial da rocha, com o espaçamento médio entre descontinuidades. Consoante os resultados obtidos na conjugação de parâmetros, é determinado se o maciço em estudo poderá ser desmontado mecanicamente ou se é necessário recorrer ao uso de explosivos.

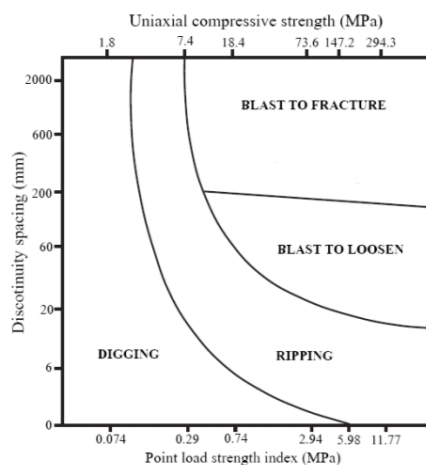


Figura 2 – Gráfico de escavabilidade de maciços rochosos (Kaya, et al., 2011)

O conhecimento prévio da litologia presente no maciço a desmontar, por exemplo, assume especial relevância no caso das explorações de minérios metálicos, onde por vezes ocorrem zonas de mudança brusca entre zonas mineralizadas e zonas onde a ganga predomina, variando também as propriedades mecânicas das rochas. Nestes casos, o desmonte terá de ser adaptado consoante o tipo de rocha, para que ocorra de forma eficiente. Jimeno (2003) aponta duas opções distintas para este tipo de desmonte:

- Dimensionar a pega de fogo de modo a que a malha seja igual, e fazendo variar as cargas em cada furo;
- Dimensionar a pega com cargas iguais por furo, mas usar malha variável – nomeadamente alterando o espaçamento entre furos.

2.3. EXPLOSIVO

Um explosivo é uma potencial fonte de energia passível de ser utilizada para promover a rotura e consequente fragmentação de um maciço rochoso por meio de uma metodologia que, no presente caso em estudo, será a execução de um desmonte em bancada. Define-se como um material que, mediante uma fonte de ignição, é capaz de produzir uma detonação, ou seja, uma libertação de energia de forma rápida, num intervalo de tempo limitado, provocando uma brusca expansão do volume da rocha. A detonação dá origem a vários fenómenos físicos, nomeadamente: geração de onda aérea, onda sonora, onda sísmica e projeção de material (Sanchidrián & Muñiz, 2000).

Distingue-se aqui o termo *detonação* do de *explosão*. O termo *detonação* contempla o máximo controlo do fenómeno de fragmentação induzida, desiderato permanentemente perseguido pelos processos de planeamento, controlo e otimização de pegadas de fogo. A palavra *explosão* tem intrínseca a noção de máxima destruição na ausência de qualquer tipo de possibilidade de controlo dos fenómenos a ela associados. Ora, nesta Dissertação, pretende-se precisamente contribuir para um aumento do processo de controlo das fenomenologias associadas à detonação através dos objetivos para ela enunciados anteriormente.

Após a invenção da dinamite houve uma natural evolução na composição dos explosivos, e atualmente há uma variada gama existente no mercado, com aplicação no desmonte de rocha. Jimeno (2003) classifica os explosivos quanto ao seu uso, emprego e velocidade de detonação, sendo classificados como explosivos militares (alto poder de fraturação e velocidades de propagação da detonação da ordem dos 6 a 9 km/s, como o TNT) ou explosivos industriais, utilizados na exploração mineira e obras civis.

Algumas das propriedades fundamentais a ter em conta no momento da escolha do explosivo a usar, são a sua potência, velocidade de detonação, densidade, pressão de detonação, estabilidade, resistência à água, sensibilidade, transmissão de detonação, resistência a baixas temperaturas e produção de gases tóxicos (Lownds, 1986).

Os explosivos podem ainda ser comercializados sob a forma de cartucho, a granel ou em emulsão, sendo estes últimos injetados ou bombeados para o interior dos furos.



Figura 3 – Explosivos encartuchados (à esquerda) e a granel (à direita)

(https://www.maxam.net/pt/terra_solutions/angola/press_room/news/bulk_technology_in_maxam_cpea)

2.3.1. EXPLOSIVOS INDUSTRIAIS

Os explosivos civis ou industriais, diferenciam-se dos explosivos militares, por serem mais sensíveis e menos estáveis, e apresentarem velocidades de detonação relativamente mais baixas, da ordem dos 2 a 8 km/s (Union Española de Explosivos, 1997).

De seguida, apresentam-se alguns tipos de explosivos industriais mais comuns e aos quais se recorre para promover a fragmentação de rocha:

- **Emulsões** - resistentes à água, potência, densidade e velocidade de detonação altas; utilizadas como carga de fundo;
- **Explosivos binários** – compostos por duas substâncias que isoladamente não são explosivas; requerem preparação no local e têm custo elevado;
- **Explosivos polvorentos** – utilizados em desmonte de rocha branda; potência, velocidade de detonação e resistência à água variáveis;
- **Explosivos secos** (ANFO e ALANFO) – baixa resistência à água, potências elevadas, densidade e velocidade de detonação baixas; bastante utilizados globalmente, principalmente como carga de coluna desmonte de rochas brandas e semiduras;
- **Explosivos de segurança** – menor potência e velocidade de detonação;
- **Gelatinas** – resistentes à água, com elevada potência, densidade e velocidade de detonação; utilizadas em desmontes de rocha muito dura e como carga de fundo.
- **Hidrogel** – muito resistentes à água, com potência e velocidade de detonação altas, utilizados como carga de fundo e em pegas de contorno.

A escolha do tipo de explosivo a utilizar é fundamental para o dimensionamento dos parâmetros geométricos de uma pega de fogo tendo em vista obter um bom resultado do desmonte. Aquando da escolha do explosivo, o projetista responsável pelo desenho do desmonte deverá ter em conta as seguintes condicionantes (Jimeno, 2003):

- Características do maciço rochoso
- Condições atmosféricas
- Condições de segurança
- Custo do explosivo
- Diâmetro de carga
- Fumos
- Presença de água
- Terrenos reativos
- Volume de rocha a desmontar

2.3.2. CONSUMO ESPECÍFICO (*POWDER FACTOR*)

Relativamente aos custos inerentes à utilização do explosivo, um dos melhores indicadores para o dimensionamento da pega de fogo é o rácio entre total de explosivo detonado num desmonte, e a quantidade de rocha libertada (kg explosivo / ton rocha), denominado consumo específico ou *powder factor*.

À medida que o *powder factor* aumenta, o grau de fragmentação diminui, para valores de afastamento constantes. Quando o afastamento é superior a 3m, o grau de fragmentação deixa de ser controlado, especialmente nas partes superiores da bancada, onde as densidades de carga são mais baixas

(Gustafsson, 1981). Usualmente o valor do consumo específico varia entre 0.1 kg/m^3 e 0.53 kg/m^3 para o desmonte em bancada.

2.3.3. MECANISMOS DE ROTURA E DISTRIBUIÇÃO DE ENERGIA

Segundo Jimeno (2003), ocorrem oito mecanismos sequenciais de rotura da rocha, quando submetida à ação do explosivo:

- Trituração de rocha
- Fissuração radial
- Reflexão da onda de choque
- Extensão e abertura de fissuras radiais
- Fraturação por libertação de carga
- Fraturação por cisalhamento
- Rotura por flexão
- Rotura por colisão

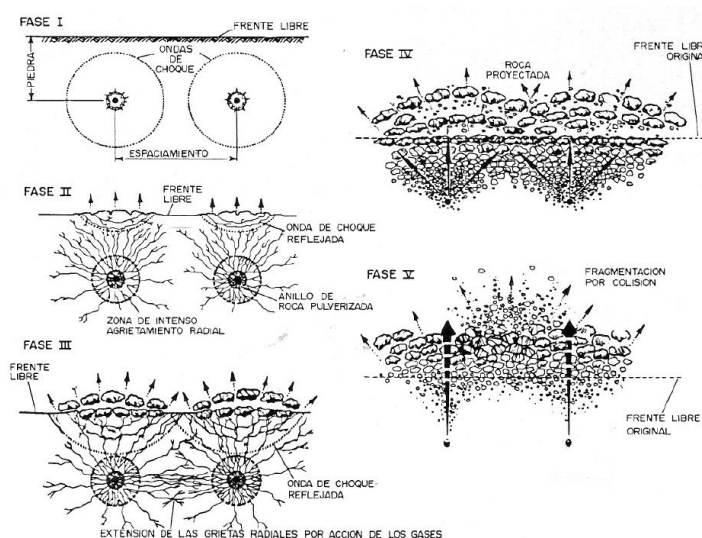


Figura 4 - Mecanismos de rotura de maciços rochosos através do uso de explosivos (Jimeno, et al., 2003)

Estes fenómenos ocorrem num curto intervalo de tempo, e são o resultado dos efeitos da onda de choque provocada pela libertação da energia do explosivo. Através da análise de um gráfico Pressão-Volume (Lownds, 1986) é possível estudar o modo como a energia do explosivo é utilizada no processo da fragmentação da rocha (Figura 5).

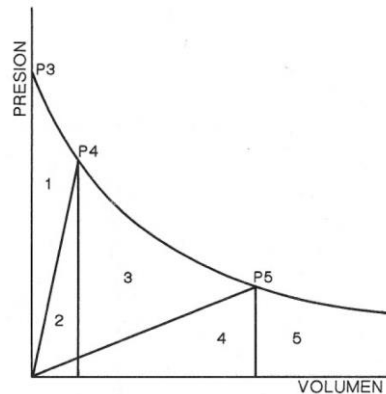


Figura 5 – Gráfico Pressão vs Volume (Jimeno, et al., 2003)

Logo após a detonação, ocorre a liberação da onda de choque provocada por gases a alta pressão (ponto P3). Esta onda de choque promove a trituração e compressão da rocha na vizinhança do furo. Assim, ocorre um aumento de volume no furo, e uma consequente redução de pressão (P4). As zonas 1 e 2 correspondem respectivamente, à componente cinética e à componente de tensão da onda de choque, sendo a soma das duas, a componente de energia de detonação. Os gases penetram nas fissuras radiais, expandindo-as até atingir o ponto de libertação (P5), em que os gases atingem a frente livre. Assim, no gráfico da Figura 5 as zonas 3 e 4 correspondem à energia libertada durante a propagação das fissuras, e as zonas 2 e 3 à energia de fragmentação.

As zonas 1,2,3 e 4 correspondem no seu todo à energia do desmonte, enquanto que a zona 5 corresponde à energia de projeção. Esta energia de projeção, quando excessiva, provoca nada mais que ruído e calor à medida que os gases se escapam, e como tal, deixa de ser energia útil.

Este gráfico P-V é importante para a compreensão do modo como a energia do explosivo é utilizada durante o mecanismo de rotura da rocha, permitindo avaliar medidas para maximizar o uso de energia útil no processo de desmonte, e minimizar a energia desperdiçada que poderá também causar efeitos indesejáveis como o ruído e as vibrações.

2.4. VARIÁVEIS GEOMÉTRICAS CONTROLÁVEIS DE UMA PEGA DE FOGO

As variáveis geométricas de uma pega de fogo são todas as variáveis que dizem respeito ao posicionamento e distribuição do explosivo relativamente à bancada e à sua frente livre. Os valores destas variáveis estão dependentes das condições geológicas e geomecânicas do maciço, e como se verificará mais adiante, são relacionáveis e até mesmo interdependentes entre si, sendo possível estabelecer relações empíricas entre os diversos parâmetros, que variam de autor para autor, sendo as relações de Ash (1963) as mais consagradas na literatura atual relativa ao desmonte de rocha, e descritas ao longo deste subcapítulo.

No decurso desta Dissertação considera-se ainda uma variável geométrica extra – a geometria da frente livre, que não é uma variável controlável, e é inerente à bancada a desmontar. Apesar de não ser uma das variáveis manipuláveis, é uma variável a ter em conta no dimensionamento da pega de fogo, uma vez que os valores de afastamento e de inclinação dos furos deverão depender da geometria da frente livre.

De seguida, apresenta-se um esquema representativo (Figura 6) com alguns dos parâmetros geométricos a ter em conta numa bancada a ser desmontada: afastamento, espaçamento, tamponamento, altura da bancada, subfuração e inclinação do furo.

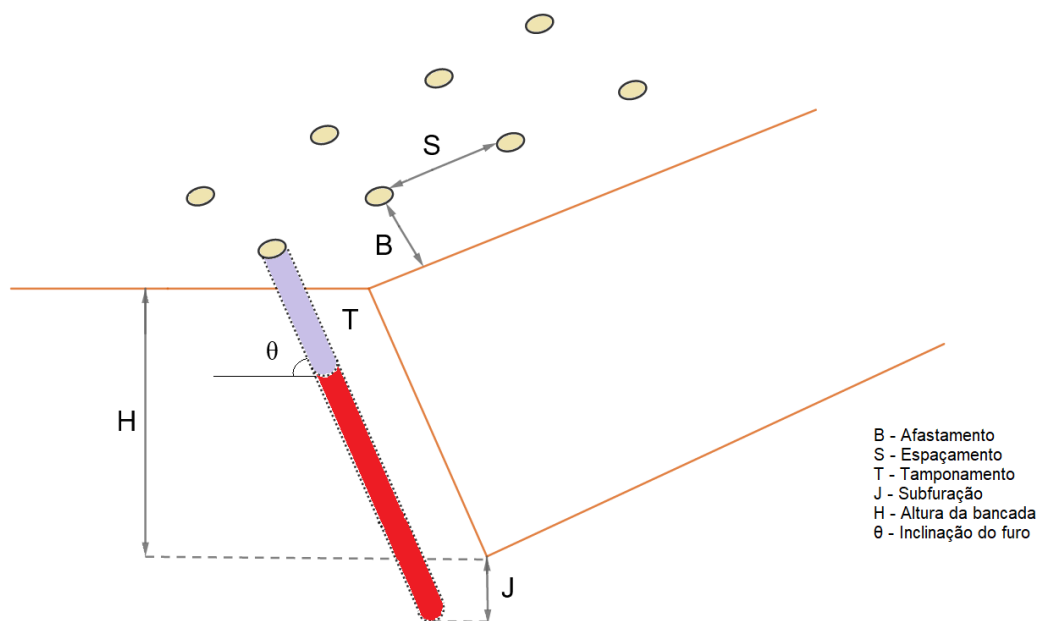


Figura 6 - Corte esquemático de bancada e furo

2.4.1. DIÂMETRO DO FURO - D

O diâmetro do furo onde irá ser introduzido o explosivo depende das características geomecânicas do maciço, do grau de fragmentação pretendido, da altura da bancada, da configuração da carga explosiva e ainda da economia dos processos de perfuração e carregamento (Jimeno, et al., 2003). Acrescenta-se aqui uma outra dependência, muitas vezes real, que se relacionará com a disponibilidade de material de perfuração inerente à entidade que promoverá o desmonte, naturalmente relacionada com a capacidade financeira da mesma. Deverá ser ainda dimensionado de acordo com o posicionamento apropriado dos furos na malha do diagrama de fogo, de modo a que a fragmentação seja adequada para os processos de carga, transporte e britagem que se seguem ao desmonte da bancada (Bhandari, 1997).

Na maioria das vezes, o aumento do diâmetro é benéfico, uma vez que permite a diminuição dos custos de perfuração e detonação, o aumento da velocidade de detonação do explosivo (estabilizando a detonação) e o aumento do rendimento dos equipamentos de carregamento do explosivo, reduzindo as zonas de baixa produtividade das pás carregadoras e dumpers, onde estes equipamentos por vezes têm problemas com a granulometria do material desmontado, que não é adequado para a sua passagem e carregamento. Contudo, é também necessário proceder ao aumento do consumo específico do explosivo, de modo a que haja um grau apropriado de fragmentação, o que dá origem a uma distribuição deficiente de cargas explosivas no maciço (Jimeno, et al., 2003).

É também relevante ter em conta que, se o espaçamento entre descontinuidades no maciço rochoso for elevado, pode ocorrer a formação de blocos, uma vez que a pressão provocada pela detonação não propaga devidamente as fissuras pelo maciço e, como tal, nestes casos recomenda-se o uso de diâmetros inferiores (Bhandari, 1997). Este autor sugere que o valor do diâmetro (D) para o qual a fragmentação atinge um valor ótimo (em mm) é de:

$$D = \frac{H}{120} \quad (\text{Eq. 1})$$

onde H é a altura da Bancada, e sendo um valor máximo admissível (em mm):

$$D_{\text{máx}} = \frac{H}{60} \quad (\text{Eq. 2})$$

Segundo Jimeno (2003) o diâmetro dos furos pode variar entre 65 mm e 450 mm, para pegas de fogo a céu aberto. Normalmente, a gama de pequenos diâmetros (65 a 165 mm) é utilizada para operações de pequena escala, de baixa produção, e a gama de grandes diâmetros (180 a 450 mm) em operações que requerem grandes produções. Bhandari (1997) é mais conservador no que toca a estes valores, assumindo que para operações de pequena escala se usem diâmetros entre 83 e 200 mm, e para operações de grande escala diâmetros entre 250 e 350 mm.

De um modo geral, num desmonte com explosivos a céu aberto, procura-se sempre usar diâmetros não muito pequenos para diminuir os custos de perfuração, e os diâmetros não variam muito entre as várias pegas de fogo ao longo da vida útil da exploração, uma vez que alterações significativas no diâmetro requerem mudanças no equipamento de perfuração e mesmo de algumas rotinas operacionais.

2.4.2. ALTURA DA BANCADA – H

A altura da bancada depende essencialmente do alcance dos equipamentos de carregamento de explosivos, e no caso de desmontes de minérios metálicos, depende também da necessidade fundamental de minimização da diluição do minério. E o seu valor é geralmente limitado. Naturalmente que questões de estabilidade dos taludes das sucessivas frentes livres geradas pelos desmontes, serão também de ter em consideração no momento da tomada de decisão do valor da altura de bancada a estabelecer.

Para alturas inferiores, qualquer variação no afastamento ou no espaçamento dos furos poderá influenciar os resultados do desmonte. Com o aumento de H, e mantendo B (afastamento do furo em relação à frente) constante, o espaçamento entre furos poderá ser aumentado sem provocar alterações na fragmentação obtida. No entanto, bancadas demasiado altas podem provocar problemas no desvio dos furos ao longo do seu comprimento, que podem afetar a fragmentação da rocha e aumentam o risco de geração de vibrações e projeções de material, dado que a malha dos furos (afastamento x espaçamento) deixa de ser constante em todas as cotas da bancada (Jimeno, et al., 2003).

Tal como o diâmetro dos furos, a altura da bancada é na grande maioria dos casos um parâmetro fixo, uma vez que depende dos limites impostos pelos equipamentos de carga e perfuração, para além de ter de assumir valores limite (geralmente da ordem dos 12 a 15 m), de modo a que as condições de segurança no local sejam cumpridas. Hustrulid (1999) sugere uma razão entre altura e afastamento, $H/B \geq 1.6$, já Bhandari (1997) sugere que essa razão seja ≥ 2.5 .

2.4.3. AFASTAMENTO – B

O afastamento é a distância medida desde o eixo de um furo até à frente livre da bancada.

Muitas vezes a frente livre é irregular, e como tal o valor do afastamento varia ao longo do comprimento do furo. Como tal, o afastamento deverá ser calculado tendo em conta as irregularidades da frente livre, de modo a que não haja excesso de energia de detonação em algumas zonas da bancada e défice de energia de detonação noutras zonas. Esta variação natural, quando associada aos desvios de perfuração provocados por erros de execução, pode trazer graves consequências para o resultado final do desmonte.

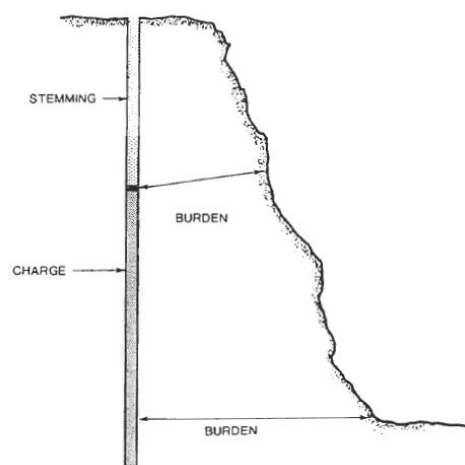


Figura 7 – Corte esquemático de um furo carregado e do perfil da frente livre (Hustrulid, 1999)

Quando o valor do afastamento é reduzido, os gases de explosão expandem e atingem a frente livre a altas velocidades, provocando a projeção descontrolada de material rochoso que poderá atingir distâncias de queda muito elevadas, eventualmente pondo em risco pessoas e bens. Quando o valor é excessivo, os gases encontram demasiada resistência da rocha à fracturação, transformando parte da energia em energia sísmica, e consequentemente provocando um aumento de vibrações (Jimeno, et al., 2003).

De acordo com Bhandari (1997) e Hustrulid (1999), o afastamento é um dos parâmetros com maior importância no dimensionamento do desmonte, condicionando o cálculo das restantes variáveis como o comprimento do furo, subperfuração, tamponamento e o espaçamento entre furos. As relações empíricas de Ash (1963) são obtidas através da conjugação destes parâmetros com o valor do afastamento.

Segundo Hustrulid (1999), a relação entre afastamento e diâmetro é dada por:

$$K_B = \frac{B}{D} \quad (\text{Eq. 3})$$

Sendo K_B função da densidade do explosivo e da rocha a desmontar. Este parâmetro varia entre 20 e 40 - valores baixos para explosivos de baixa densidade, e valores altos para explosivos de alta densidade.

Conhecendo K_B , é possível obter o valor do afastamento B , em função deste parâmetro e do diâmetro dos furos, que não deverá variar.

2.4.4. INCLINAÇÃO DOS FUROS

A inclinação dos furos vai naturalmente influenciar o afastamento a cada profundidade ao longo do furo. No limite, a inclinação do furo deveria permitir que o afastamento do furo à frente livre permanecesse constante. Como na prática, tal situação é bastante difícil de atingir dadas as irregularidades da frente, optando por realizar furos inclinados, a perfuração deverá garantir que o furo seja o mais paralelo possível à frente livre. Assim, ao longo de todo o comprimento do furo, o afastamento pode variar bastante, consoante as características geométricas da frente livre, sendo que em algumas zonas, esses valores de afastamento poderão ser valores críticos.

A prática de implementação de furos inclinados permite, entre outras vantagens: melhor fragmentação do material, frentes mais regulares e seguras após o desmonte, melhor reflexão da onda de choque no pé da bancada permitindo diminuir o consumo específico de explosivo, diminuição das vibrações, e maiores rendimentos dos equipamentos de perfuração e carregamento de material. A todas estas vantagens, a possibilidade de escolha da inclinação dos furos acrescentará a vantagem da minimização de afastamentos críticos, se não mesmo a sua eliminação.

Contudo, tem como principais inconvenientes o potencial desvio dos furos, requerendo uma supervisão mais atenta das atividades dos operadores das máquinas de perfuração, problemas de carregamento quando o furo tem água e um aumento de ocorrência de projecções se a carga de explosivo estiver sobredimensionada.

Habitualmente, a inclinação varia entre 0° e 20° (Bhandari, 1997).

2.4.5. TAMPONAMENTO – T

O tamponamento (T), também conhecido como atacamento (*stemming* no inglês), é a variável que diz respeito à parte superior do comprimento do furo que é preenchida com material inerte, geralmente detritos resultantes da operação de perfuração, sendo o seu objetivo o confinamento dos gases produzidos pelo explosivo, de modo a que estes não se escapem e assim aumente a eficácia de fraturação e deslocamento apropriados do material rochoso (Bhandari, 1997). Bhandari estabelece a seguinte relação entre o comprimento do tamponamento e o afastamento:

$$K_T = \frac{T}{B} \quad (\text{Eq. 4})$$

Sendo que o valor de K_T deverá situar-se entre 0.7 e 1.5.

Jimeno (2003) relaciona também o valor do tamponamento com o do diâmetro do furo, devendo assumir valores entre 20D e 60D – valores abaixo desta gama provocam fuga prematura de gases de explosão e podem gerar problemas de onda aérea e projeções; valores acima desta gama promovem a formação de blocos na zona alta da bancada.

2.4.6. SUBFURAÇÃO – J

A subfuração (J) é a perfuração “extra” realizada na zona inferior do comprimento do furo, situada abaixo do nível do pé da bancada. Este parâmetro deverá garantir o desmonte total da frente livre, originando um novo piso inferior da bancada regular e sem o conhecido efeito de repé. Naturalmente que, quando o valor de J é insuficiente, ocorre a geração de repés que se traduzirão em irregularidades do piso inferior da bancada que afetarão significativamente a movimentação das máquinas de carga e transporte do material desmontado, podendo provocar sucessivos efeitos de repés, que dêem origem a pisos com inclinações mais ou menos acentuadas. Quando este comprimento de subfuração é excessivo, ocorre um aumento de vibrações e fragmentação excessiva no pé da bancada.

Ash (1963) define uma relação entre a subfuração e o afastamento:

$$K_J = \frac{J}{B} \quad (\text{Eq. 5})$$

Em que J/B deverá ser da ordem do valor 0.3 a 0.5, e como tal, o valor de comprimento de subfuração deverá ser de 30% a 50% do valor do afastamento (Bhandari, 1997).

2.4.7. ESPAÇAMENTO – S

A variável denominada espaçamento (S), em conjunto com o afastamento, controla o dimensionamento da malha do diagrama de fogo. É a distância entre furos, medida perpendicularmente ao afastamento. Calcula-se em função do afastamento, do comprimento do furo e dos tempos de iniciação (Bhandari, 1997).

De acordo com Bhandari (1997), Hustrulid (1999) e Jimeno (2003), o espaçamento depende do diâmetro, das propriedades das rochas e dos explosivos, da altura da bancada e do grau de fragmentação desejado, mas depende de forma particular do afastamento, podendo estabelecer-se a seguinte relação ($K_s = S/B$) entre ambos os parâmetros:

$$1 \leq \frac{S}{B} \leq 2 \quad (\text{Eq. 6})$$

- $S/B = 1$, em pegas de fogo com elevados tempos de retardo;
- $S/B = 2$, em pegas de fogo em que as cargas são iniciadas simultaneamente;
- S/B está entre 1 e 2, em pegas de fogo com tempos de retardo curtos.

2.5. ACESSÓRIOS DE INICIAÇÃO E DETONAÇÃO

A detonação do explosivo decorre após a sua iniciação mediante uma fonte de ignição – detonador.

Uma vez que o detonador, por si só, não deverá fornecer energia suficiente para a detonação da carga explosiva contida no furo, principalmente quando são utilizados explosivos pouco sensíveis ou insensíveis, torna-se por vezes necessária a utilização de um acessório que seja capaz de amplificar esta energia, promovendo a correta detonação da carga do furo. Normalmente são utilizados *boosters* (multiplicadores) à base de pentolite (mistura de PETN com TNT). Este tipo de auxiliar, produz altas velocidades e pressões de detonação, garantindo assim uma iniciação ótima dos agentes explosivos, tendo como grande vantagem o facto de serem seguros, resistentes à água e a condições de armazenamento extremas (Union Española de Explosivos, 1997).



Figura 8 - *Booster* (http://www.explosivos.com/?page_id=13865)

Atualmente, existem três tipos de detonadores utilizados em explorações mineiras: elétricos, não-elétricos e eletrônicos. Estas três soluções dispõem de detonadores instantâneos e de detonadores com tempos de retardo. A combinação dos tempos de retardo é crucial para o bom desempenho do desmonte, uma vez que permite controlar a direção de projeção do material, a fragmentação e o empilhamento, mas também controlar o número de furos que detonam simultaneamente (que deverá ser o menor possível para evitar a geração de vibrações).

2.5.1. DETONADORES ELÉTRICOS

Os detonadores elétricos, como o próprio nome indica, funcionam com recurso à energia elétrica. São artefactos iniciadores da detonação seguros e que permitem obter muito bons resultados. O seu modo de funcionamento consiste em fazer passar uma corrente elétrica num fio de conexão com uma resistência rodeada de material (pasta) explosivo. Com a passagem de corrente, a resistência aquece, até ao ponto em que provoca reação na pasta explosiva. No caso dos detonadores instantâneos, esta reação é imediata. Contudo, no caso dos detonadores com tempo de retardo, esta reação é atrasada pela presença de uma carga retardadora (Union Española de Explosivos, 1997).

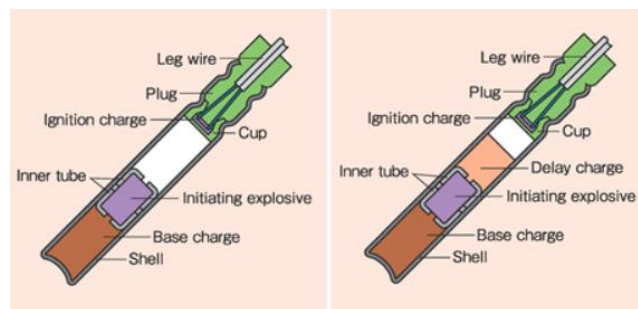


Figura 9 – Corte esquemático de um detonador elétrico instantâneo (à esquerda) e com retardo (à direita)
(<http://www.kayakujapan.co.jp/en/sale/denkiraikan.html>)

Estes acessórios classificam-se consoante:

- A sua sensibilidade – sensíveis, insensíveis e altamente insensíveis.
- O uso – resistentes a alta pressão de água para pegas de fogo submarinas, detonadores de cobre para ambientes com presença de grisú e detonadores para prospeção sísmica.
- Os tempos de detonação – instantâneos ou temporizados (com tempo de retardo variável entre 20 ms e 500 ms)



Figura 10 – Aspeto exterior de um detonador elétrico
(http://www.oricaminingservices.com/ee/en/product/products_and_services/initiating_systems/page_initiating_systems/electric_detonator_sd/1385)

Relativamente à classificação quanto à sensibilidade:

- Os detonadores sensíveis são adequados para locais sem ocorrência de correntes estáticas provocadas por linhas elétricas ou tempestades;
- Os detonadores insensíveis são utilizados nos casos onde não é previsível a ocorrência de energia elétrica perigosa;
- Os detonadores altamente insensíveis são os mais seguros e apropriados para os ambientes onde há ocorrência de energia elétrica perigosa, que possa despoletar a auto-iniciação do detonador.

Numa pega de fogo os detonadores estão conectados entre si, formando um circuito unido a uma fonte de energia. É possível conectar os detonadores usando:

- Circuito em série, simples e mais fácil de verificar a continuidade do circuito
- Circuito em paralelo, usado sobretudo em trabalhos subterrâneos e submarinos
- Circuito misto, quando o número de detonadores é muito grande e é necessário reduzir a resistência total

2.5.2. DETONADORES NÃO-ELÉTRICOS

O princípio de ignição destes detonadores baseia-se numa onda de choque com baixa velocidade de detonação que atravessa um tubo de transmissão. O tubo, feito de plástico, tem um diâmetro de 3mm e possui uma película de material reativo (cerca de 15 gramas) na sua face interior, que quando iniciado, origina uma onda de detonação de baixa energia a uma velocidade de 2100 m/s (Dyno Nobel, 2011).

Têm como grande vantagem, o facto de poderem ser utilizadas combinações de tempos ilimitadas, e podem ser utilizados com todo o tipo de explosivos, para além de que não apresentam risco de auto-detonação na presença de fontes de energia elétrica (Union Española de Explosivos, 1997). Estes detonadores são ligados entre si através de conectores, alguns deles com soluções para incorporação do tubo de choque e até mesmo de cordão detonante.



Figura 11 – Detonador não-elétrico (à esquerda) e conector (à direita) (Dyno Nobel, 2011)

Os tempos de iniciação com um detonador não-elétrico, e o modo como as conexões entre furos são feitas, influenciam a trajetória do material rochoso, como é visível nos exemplos das figuras Figura 12 e Figura 13. A iniciação dá-se a partir do furo-piloto, e as cargas vão sendo detonadas sequencialmente, de modo a que o material seja fragmentado e projetado na direção pretendida. É função do projetista (por vezes com um pouco de criatividade) providenciar a melhor solução que combine os tempos de retardo e as conexões (entre filas ou em diagonal), com a direção pretendida para a saída do material.

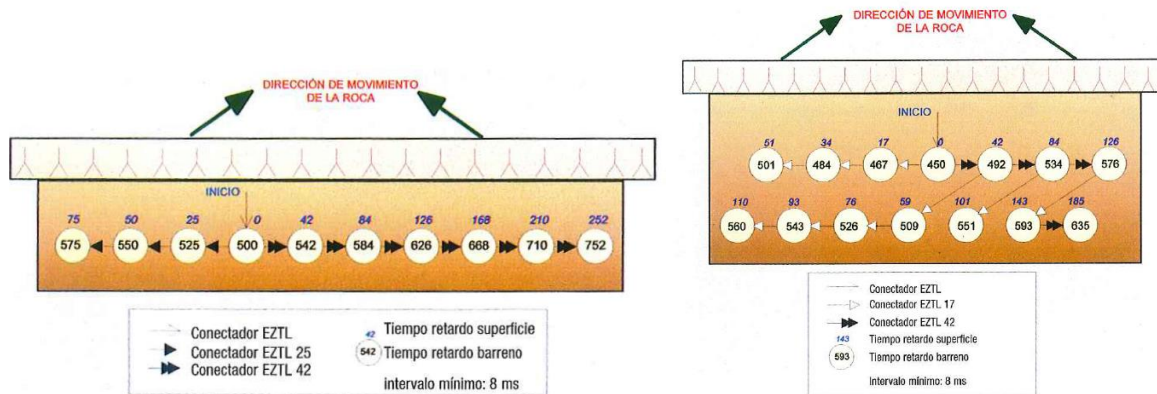


Figura 12 – Pegas de fogo com saída do material para o centro da bancada (Union Española de Explosivos, 1997)

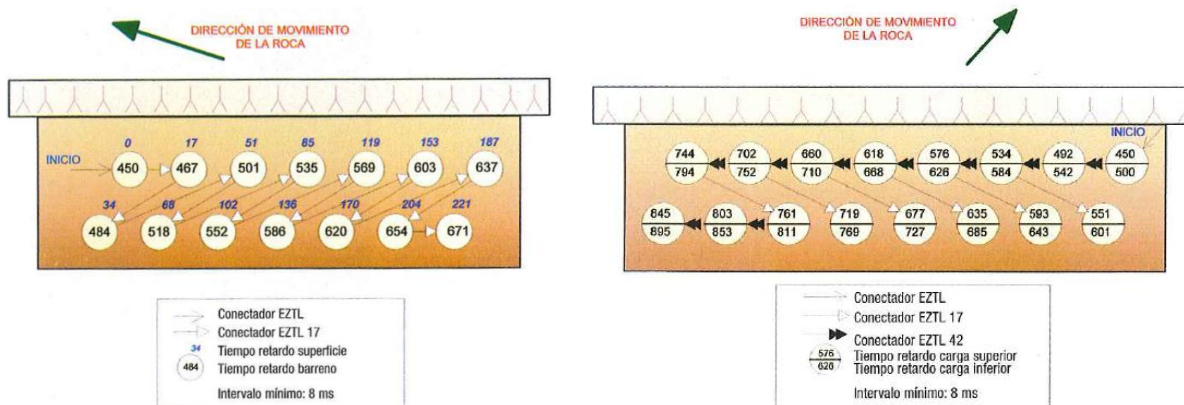


Figura 13 – Pegas de fogo com saída do material para as laterais da bancada (Union Española de Explosivos, 1997)

2.5.3. DETONADORES ELETRÓNICOS

O detonador está conectado a uma unidade central (computador) que codifica um sinal próprio para cada detonador, permitindo controlar individualmente os tempos de iniciação de cada detonador envolvido, possibilitando um número muito maior de opções para escolha dos tempos de retardo e uma combinação infindável de tempos entre furos (Jimeno, 2003). Consiste numa cápsula com carga primária, carga de iniciação, um microchip e um capacitor.



Figura 14 – Unidade central (computador) de um sistema eletrônico de detonação
(<http://www.detnet.com/products/underground-mining/digishot-plus>)

Permitem um controlo preciso da fragmentação, redução de projeções de material e vibrações, e extremamente seguros em ambientes sob tensão elétrica. Têm como principal desvantagem o facto de serem bastante mais caros que as outras opções do mercado, e dos operadores que os manuseiam tenham de se submeter a treino algo intensivo (Cardu, et al., 2013). O facto de ser uma tecnologia recente com elevado custo, tem levado a que este tipo de detonador seja preterido em relação aos elétricos e não-elétricos.

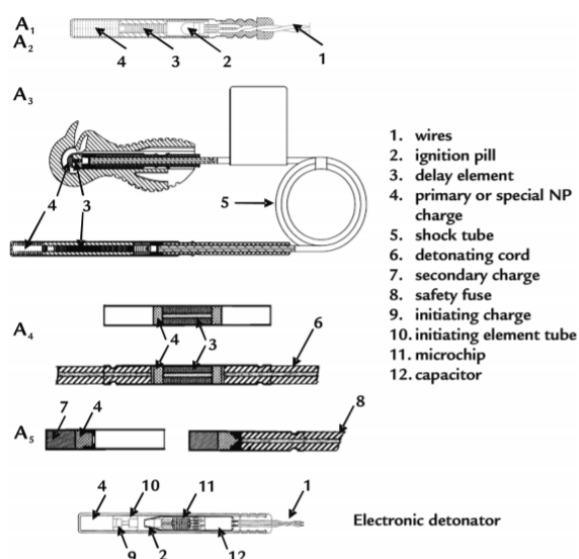


Figura 15 – Comparação entre sistemas elétricos e não-elétricos, com o sistema eletrônico (Cardu, et al., 2013)

3

FUNDAMENTOS DE PROGRAMAÇÃO LINEAR

3.1. INVESTIGAÇÃO OPERACIONAL

3.1.1. RESENHA HISTÓRICA

A Investigação Operacional é uma das áreas mais importantes e transversal a todos os cursos de Engenharia, sendo uma ferramenta fundamental no apoio à tomada de decisões (Taha, 2008), e que de acordo com Dantzig, o criador do método SIMPLEX¹, na sua entrevista ao *The College Mathematical Journal* (1986), recorre a algoritmos e modelos matemáticos para modelar e solucionar uma grande quantidade de problemas de forma óptima.

Embora as primeiras referências a esta área remetam para o séc. XVI, foi no decurso da 2ª Guerra Mundial que a Investigação Operacional foi verdadeiramente impulsionada, graças aos esforços dos Aliados para desenvolver a distribuição logística de recursos de maneira óptima, cujo objetivo era promover estratégias militares que levassem à vitória nos conflitos (Gass & Assad, 2005).

O grande despontar dos métodos de Investigação Operacional deveu-se a Patrick Blackett (1940) e ao seu grupo de trabalho, através do estudo dos sistemas de defesa anti-aérea controlados por radar, tendo identificado ao longo das suas observações alguns problemas relativamente à precisão do bombardeamento de submarinos. De acordo com Tormos Juan *et al.* (2003), este estudo englobou algumas etapas-base para o que viria a ser a Investigação Operacional moderna:

1. Aquisição de dados
2. Modelação matemática do problema
3. Obtenção da solução-óptima para o problema
4. Modificação da solução de acordo com factores reais não-considerados pelo modelo – a solução de Blackett para melhorar a precisão do bombardeamento de submarinos levou à fabricação de espoletas que detonavam a 10m de profundidade, que não eram fabricadas até à data.

Com o final da 2ª Guerra Mundial, a Investigação Operacional foi sendo rapidamente implementada para fins civis, dada a sua enorme utilidade para a resolução de problemas típicos da indústria, negócios e administração pública. Esta área progrediu bastante logo após o término da guerra, tendo já ficado estabelecidas algumas das técnicas utilizadas ainda nos dias que correm, como é exemplo o

¹ O método de resolução de problemas de programação linear SIMPLEX é abordado pedagogicamente no excelente terceiro capítulo de “*Pesquisa Operacional*” (Taha, 2008)

desenvolvimento do algoritmo Simplex para resolução de problemas de programação linear por Dantzig (1947) e outras áreas como a Programação Dinâmica, Teoria das Filas e Teoria de Inventários (Tormos Juan & Lova Ruiz, 2003). A evolução tecnológica e a massificação do uso de computadores, promoveu também o grande desenvolvimento e aplicação generalizada das técnicas de Investigação Operacional, graças ao poder de resolução de problemas complexos através de computadores (Hillier, 2010).

Pode-se destacar, de acordo com a bibliografia atual, oito tipos distintos de técnicas aplicadas na Investigação Operacional:

- Controlo de Inventários;
- Teoria de filas;
- Fiabilidade de máquinas e equipamentos;
- Teoria de jogos;
- Modelos de otimização de redes;
- Gestão de Projetos (PERT/CPM);
- Simulação;
- Programação matemática;
- Processo combinado - uso de mais de uma das técnicas de I.O.

3.1.2. FASES E TÉCNICAS DA INVESTIGAÇÃO OPERACIONAL

Os métodos de Investigação Operacional seguem uma lógica semelhante à do método científico, nomeadamente no que concerne a definição do problema a ser estudado. Hillier (2010), define seis etapas-base para a construção do estudo de Investigação Operacional:

1. Definição do problema e recolha de dados;
2. Construir um modelo matemático representativo do problema;
3. Resolução do modelo de forma ótima (com recurso a ferramentas informáticas);
4. Testar e rever o modelo, se necessário;
5. Preparação do modelo para a sua implementação;
6. Implementação do modelo.

Todas elas extremamente importantes para que o estudo seja bem-sucedido, mas com especial ênfase para a definição do problema, uma vez que é imperativo trabalhar com uma definição correta e precisa do problema, de modo a obter um modelo e soluções válidas, que se adequem à realidade do fenómeno em estudo. Igualmente importante, é a construção do modelo matemático, uma vez que a escolha do modelo afeta a obtenção de uma solução-ótima capaz de satisfazer o problema (Tormos Juan & Lova Ruiz, 2003).

3.1.3. MODELAÇÃO MATEMÁTICA

Um modelo matemático é uma representação abstrata de algo existente na realidade, ou seja, é uma tradução de um fenómeno ou problema para a linguagem matemática.

Na construção de um modelo matemático, é importante a definição de três conceitos (Tormos Juan & Lova Ruiz, 2003): variáveis de decisão, restrições, e a função objetivo que se pretende otimizar.

3.1.3.1. Variáveis de decisão, restrições e função objetivo

As variáveis de decisão são as variáveis suscetíveis de serem controladas num dado processo ou fenómeno. Na resolução do modelo, estas variáveis dizem respeito às incógnitas cujas soluções se pretendem determinar, de modo a resolver o problema de forma ótima.

As restrições são introduzidas no modelo de modo a que sejam tidas em conta as limitações físicas do problema, e servem para restringir as variáveis de decisão de modo a que as suas soluções estejam contidas num certo intervalo de valores que respeite a natureza do problema. São habitualmente expressas por equações ou inequações.

A função objetivo tem como finalidade definir a solução ótima para o problema em estudo, ou seja, o valor máximo ou mínimo (consoante o tipo de problema).

3.1.3.2. Tipos de modelo matemático

O modelo matemático deve ser construído com base na natureza do problema, e como tal podem definir-se alguns tipos de modelo distintos, consoante as propriedades que o modelo apresenta na realidade. Como tal, existem três tipos de modelos distintos: lineares, mistos ou não-lineares.

3.2. PROGRAMAÇÃO LINEAR

Os modelos lineares são aqueles onde a função objetivo e as restrições são lineares, e as variáveis de decisão contínuas. Os modelos lineares obtêm um ótimo global, sendo o algoritmo Simplex o mais comumente utilizado (Tormos Juan & Lova Ruiz, 2003).

Um modelo de programação linear pode ser definido da seguinte forma (Hillier, 2010):

$$\text{máx ou mín } (Z) = c_1X_1 + c_2X_2 + c_3X_3 + \dots + c_n X_n \quad (\text{Eq. 7})$$

Onde as variáveis X_i correspondem às variáveis de decisão, c_j aos coeficientes da função objetivo (ou coeficientes tecnológicos) e Z a função a otimizar. Esta função objetivo estará sujeita às restrições que se formulam da seguinte forma, tendo em conta o tipo de limitação de cada restrição:

$$a_{m1}X_1 + a_{m2}X_2 + a_{m3}X_3 + \dots + a_{mn}X_n \geq b_m \quad (\text{Eq. 8})$$

ou

$$a_{m1}X_1 + a_{m2}X_2 + a_{m3}X_3 + \dots + a_{mn}X_n = b_m \quad (\text{Eq. 9})$$

ou

$$a_{m1}X_1 + a_{m2}X_2 + a_{m3}X_3 + \dots + a_{mn}X_n \leq b_m \quad (\text{Eq. 10})$$

Sendo a_{ij} os coeficientes técnicos, b_i o valor que representa a disponibilidade de um recurso (limitação), n o número de variáveis de decisão e m o número de restrições.

3.2.1. SOLUÇÃO ÓTIMA

Em programação linear, existe um conceito ambíguo de solução – uma solução pode ser viável (caso cumpra todas as restrições) ou não-viável (caso viole pelo menos uma das restrições).

Uma solução ótima será aquela que, para além de ser uma solução viável, deverá corresponder ao maior valor das soluções, caso se pretenda maximizar a função objetivo, ou ao menor valor das soluções, caso se pretenda minimizar a função objetivo (Taha, 2008). Poderá ainda dar-se o caso do problema ter uma infinidade de soluções ótimas (Figura 16), ou de não ter qualquer solução ótima (Figura 17).

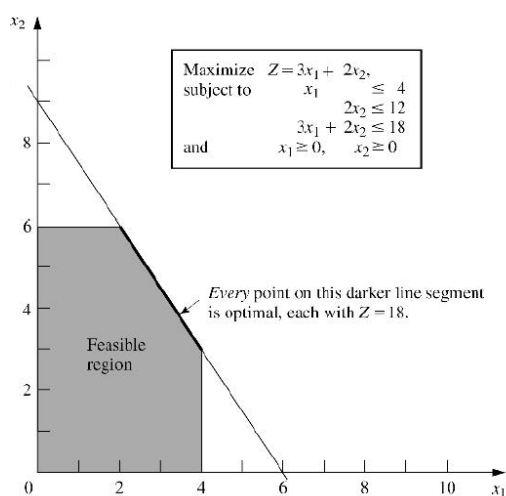


Figura 16 – Problema com múltiplas soluções ótimas (Hillier, 2010)

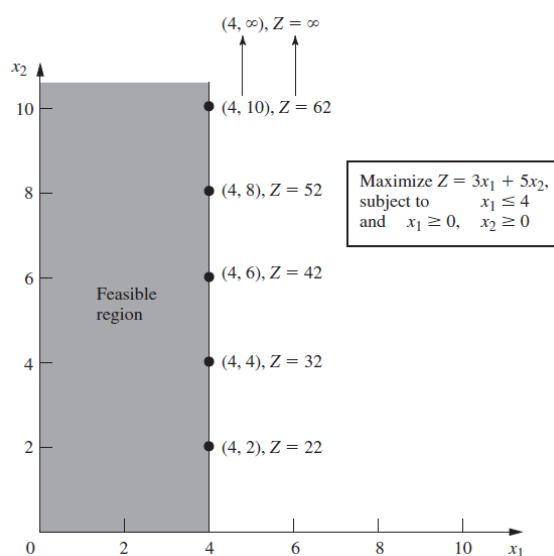


Figura 17 – Problema sem solução ótima (Hillier, 2010)

3.2.2. MÉTODO SIMPLEX

Como já foi mencionado em 3.1., este método foi desenvolvido logo após a 2ª Guerra Mundial, por George Dantzig, em 1947. É um algoritmo bastante poderoso para a resolução de problemas lineares complexos, principalmente quando é utilizado com recurso a ferramentas informáticas.

O Simplex, à semelhança de outros algoritmos e técnicas de Investigação Operacional, tem uso no quotidiano para uma quantidade infindável de problemas em diversas áreas como a agricultura, engenharia, gestão, nutrição ou transportes.

3.2.2.1. Solver do Microsoft Excel

Apesar de existirem bastantes *softwares* dedicados à resolução de problemas de otimização, atualmente em meio académico e não só, a maioria destes problemas podem ser resolvidos com recurso à ferramenta *Solver* do Microsoft Excel.

Basta para tal, proceder à formulação do problema numa folha de cálculo, atribuindo às diferentes células os valores de função objetivo, restrições e variáveis de decisão. O problema deverá ficar bem definido na folha de cálculo, de modo a que possa ser utilizada eficazmente a ferramenta *Solver*. De seguida procede-se à definição da função objetivo (indicando se é um problema de maximização, minimização ou procura de valor), das variáveis de decisão a serem alteradas pelo *Solver*, e das restrições do problema.

Esta ferramenta possui ainda três tipos de métodos de resolução de problemas de optimização:

- LP Simplex – resolução de problemas lineares;
- GRG Não-Linear – resolução de problemas não-lineares;
- Evolutionary – resolução de problemas não-uniformes.

Consoante as soluções obtidas, é ainda possível fazer uma análise de sensibilidade no Excel, de modo a observar o efeito da variação de algum dos parâmetros na solução obtida, podendo fazer ajustes ao modelo *a posteriori*.

3.2.2.2. Função *linprog* do MATLAB

O MATLAB dispõe de uma *toolbox* de otimização com recurso a programação linear, quadrática, não-linear. No âmbito da programação linear, esta caixa de ferramentas possui uma função denominada *linprog* que é dedicada à resolução de problemas lineares, encontrando um mínimo para o problema em questão:

$$\min_x f^T x \mid A \cdot x \leq b \quad (\text{Eq. 11})$$

onde f , x e b são vetores e A uma matriz. Esta minimização, em termos de linguagem MATLAB traduz-se num código do tipo $X = \text{linprog}(f, A, b)$.

A variável de saída x corresponde ao vetor x definido na equação 11 e é, portanto, um vetor que contém a solução do problema.

Adicionalmente, é possível fazer o código devolver um valor que descreve as condições de saída e a estrutura que contém a informação sobre o processo de otimização, usando para tal efeito o comando $[x, fval, exitflag, output] = \text{linprog}(f, A, b)$.

O parâmetro de saída *exitflag* pode variar entre os valores {1, 0, -2, -3, -4, -5, -7} consoante a condição de saída da função, sendo que o valor de *exitflag* = 1 é o valor que indica que o programa converge para uma solução, *exitflag* = 0 o valor que indica que o número máximo de iterações foi atingido, e os restantes valores negativos de *exitflag* indicam que o programa não encontrou uma solução por diversos problemas que o operador deverá tentar corrigir, se for possível.

O parâmetro de saída *output* dá informação sobre o processo de otimização, devolvendo uma estrutura com os seguintes dados:

- número de iterações (*iterations*);
- algoritmo de otimização utilizado (*algorithm*);
- mensagem de saída (*message*);
- máximo das funções de restrição (*constrviolation*);
- medida optimalidade de primeira ordem (*firstorderopt*).

No que toca à formulação do problema, os parâmetros de entrada *f*, *A*, e *b*, deverão estar corretamente definidos e da seguinte forma:

- *f* é um vetor que representa a função objetivo $f^T x$. O vetor *f* deverá ser um vetor-coluna de dimensão $n \times 1$, mas caso seja feito o *input* com um vetor-linha, a função *linprog* faz a transposição do vetor automaticamente;
- *A* é uma matriz de dimensão $m \times n$, onde *m* é o número de restrições do problema e *n* é o número de variáveis do problema (coincidente com o comprimento de *f* e *x*) e é constituída pelos coeficientes associados às restrições do problema;
- *x* é um vetor-coluna de dimensão *n*, representativo das variáveis do problema. Não é feito o *input* deste vetor no programa;
- *b* é um vetor-coluna de dimensão $m \times 1$, que representa os valores-limite das restrições do problema.

Em suma, os parâmetros deverão estar definidos de acordo com a inequação (Eq. 10) observada anteriormente:

$$a_{m1}X_1 + a_{m2}X_2 + a_{m3}X_3 + \dots + a_{mn}X_n \leq b_m$$

4 MODELAÇÃO

4.1. DEFINIÇÃO DO PROBLEMA

Como foi possível observar no capítulo 2 do presente trabalho, um dos principais problemas num desmonte a céu aberto com recurso a explosivos, prende-se com a conjugação ótima de parâmetros do diagrama de fogo e na adequação da geometria do plano de fogo e da carga ao terreno que se pretende desmontar (Miranda, et al., 2014).

O objetivo de uma pega de fogo deverá passar por obter o melhor desmonte com a menor geração de efeitos indesejáveis (projeções, ruído, vibrações), menor custo, e se possível libertando novas frentes de bancada o mais regulares possível, de modo a que as pegas de fogo seguintes tenham também malhas mais regulares.

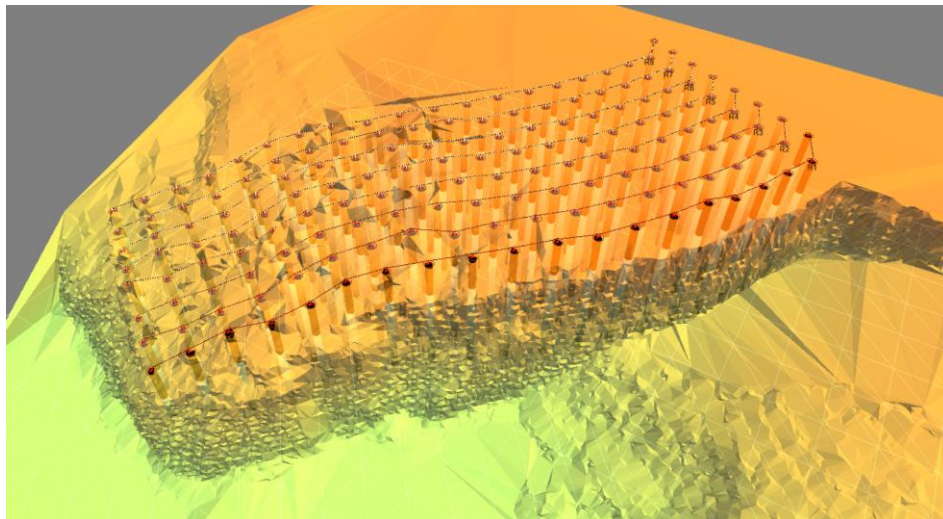


Figura 18 – Bancada modelada na plataforma O-Pitblast®

Na realidade, quando o afastamento à frente livre varia ao longo do furo, ocorrem problemas de distribuição de carga. Nas zonas do furo mais próximas da frente livre ocorre geração de energia excessiva, provocando fragmentação excessiva, enquanto nas zonas do furo mais afastadas, há ausência de energia adequada, provocando a geração de blocos. Como tal, o projetista terá de considerar a redistribuição de carga ao longo do furo para tentar minimizar estes efeitos. Considerando uma bancada com frente irregular, a malha de fogo deverá ser dimensionada tendo em atenção essa irregularidade.

A modelação com recurso a *software*, como o caso do O-Pitblast®, constitui uma ferramenta bastante poderosa e de grande auxílio para o projetista encarregue de desenhar a pega de fogo. No entanto, alguns dos parâmetros relacionados com o dimensionamento dos furos, têm o inconveniente de terem de ser ajustados manualmente pelo operador, como é o caso da inclinação do furo. Apesar de o *software* indicar qual o afastamento crítico do furo em relação à frente livre, terá de ser o projetista a fazer variar a inclinação, ou outros parâmetros, de modo a que o furo seja dimensionado da melhor forma, respeitando sempre os valores limite para cada variável, e maximizar a homogeneidade da distribuição dos afastamentos ao longo do furo. Este inconveniente é visível nas figuras Figura 19 e Figura 20, em que o utilizador ajusta a inclinação do furo manualmente, e ainda assim o furo não cumpre o valor do afastamento crítico desejável.

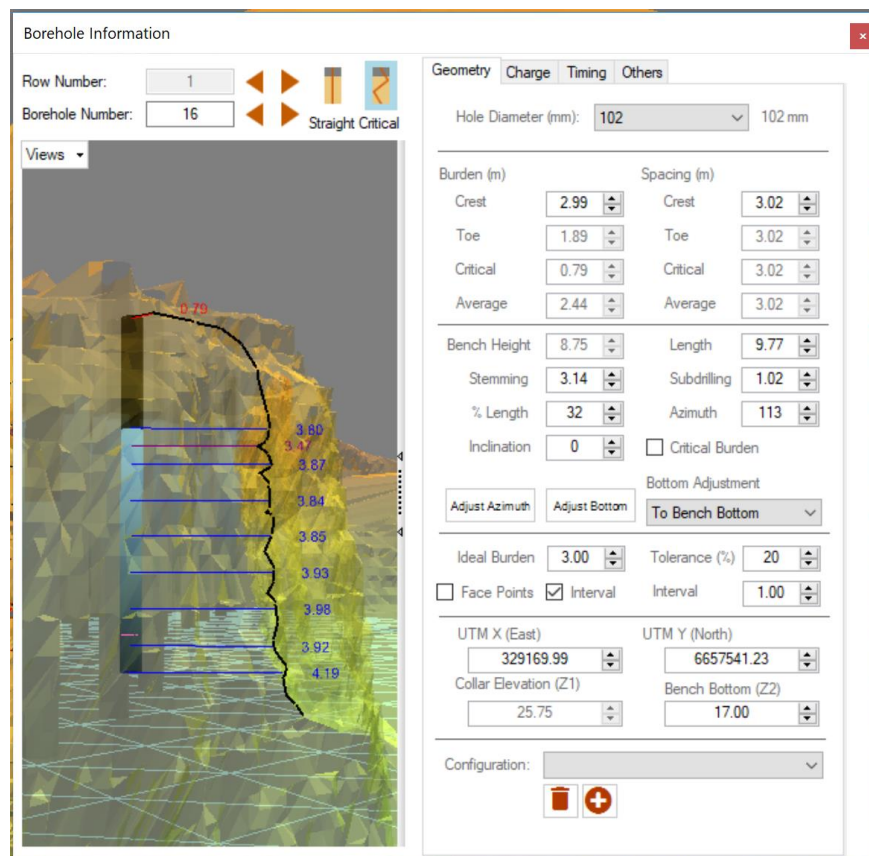


Figura 19 – Modelo de furo vertical gerado na plataforma O-Pitblast®

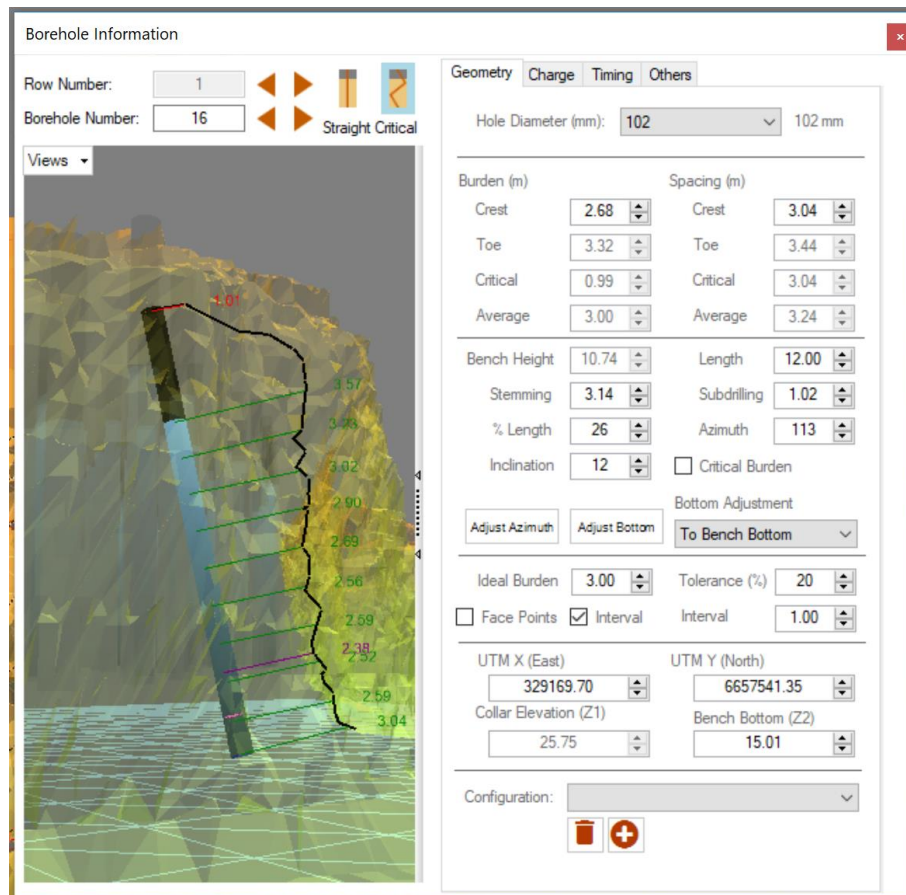


Figura 20 – Modelo de furo ajustado à frente livre

Assim, torna-se útil uma ferramenta que seja capaz de, dada uma determinada configuração geométrica da frente livre, calcular automaticamente qual a posição do furo, a sua inclinação, o valor do afastamento crítico, e outros parâmetros indispensáveis ao correto dimensionamento da pega de fogo.

Este processo de otimização pode ser obtido através da construção de um algoritmo baseado nos fundamentos de programação linear, revistos no capítulo 3 da presente Dissertação, que seja capaz de ajustar automaticamente a inclinação de um furo, dado um *input* correspondente apenas a um conjunto de pontos representativos da frente livre, e a uma estimativa para o afastamento ideal do furo em relação à frente livre. O algoritmo deve ser capaz de otimizar a inclinação do furo, respeitando sempre o valor do afastamento crítico, e tentando manter um valor do afastamento em relação à frente o mais uniforme possível para evitar a má distribuição de carga explosiva ao longo do furo.

4.2. TRATAMENTO DE DADOS EM MATLAB

O primeiro passo na abordagem deste problema, está relacionado com o tratamento dos dados adquiridos previamente.

Por vezes, mesmo após um tratamento prévio dos dados brutos obtidos *in situ*, os dados poderão não estar ainda preparados para serem trabalhados de modo eficaz na construção do modelo, e como tal, justifica-se a construção de uma ferramenta que seja capaz de tratar os dados. Uma ferramenta adequada para este tipo de problema, deverá ser capaz de filtrar *outliers*, e ainda manipular os dados de modo a que sejam mais leves de processar pelo computador, e ainda assim sejam devidamente representativos da amostra que foi recolhida no terreno.

No anexo A.1. está presente um algoritmo, em linguagem MATLAB, que diz respeito ao tratamento de *outliers* e posterior resolução do problema através de programação linear, fazendo a otimização da inclinação dos furos relativamente à frente livre. Parte deste código destina-se ao tratamento dos dados recolhidos por meio de um aparelho como é exemplo o perfilómetro laser, para posteriormente ser feita a modelação e otimização linear do problema de modo mais eficiente, já com os *outliers* excluídos.

4.2.1. REPRESENTAÇÃO GEOMÉTRICA DO TERRENO E REDUÇÃO DA AMOSTRA

A primeira abordagem feita pelo algoritmo é a importação dos dados relativos ao terreno, obtidos *in situ*, através de instrumentos de levantamento topográfico.

A importação dos pontos representativos do terreno, é feita através de um *input* das coordenadas cartesianas da frente livre, sob a forma de ficheiro de folha de cálculo (neste caso, denominado *terrain.csv*), e o *output* final é a representação bidimensional do afastamento crítico correspondente a um perfil crítico da frente livre, já sem *outliers*.

```
14 % Leitura dos dados
15
16 - data=csvread('terrain.csv',1,0); %dados de terreno
17 - collar=csvread('holes.csv',1,0); %dados dos furos
18 - toe = collar(:,4:6); % colunas relativas aos pés dos furos
19 - collar = collar(:,1:3); %colunas relativas às cristas dos furos
20
21 % ATENÇÃO -> o arquivo pode ser substituído para novo terreno ou novos
22 % furos (basta mudar nome dos ficheiros)
```

Figura 21 - Obtenção de dados de terreno e de furos

As coordenadas da frente livre correspondem graficamente a uma nuvem com um determinado número de pontos. É natural que, no que concerne o tratamento de dados, o número de pontos obtido seja excessivo, uma vez que um *software* como o MATLAB, apesar de ser uma ferramenta potente, tem obviamente as suas limitações e depende também do poder de processamento do computador em que está instalado. Como tal, é necessário reduzir a nuvem de pontos original a uma amostra que seja representativa da frente livre.

Para a redução de amostra, considerou-se um número máximo de 1000 pontos. Numa mesma etapa, foram ainda definidos os valores dos parâmetros de afastamento crítico e de tamponamento (ambos considerados com valor de 3 metros), de modo a que seja possível a obtenção dos furos respeitando estas condições, como será visto mais adiante.

```

24 % Input de variáveis
25
26 - maxPoints=1000; % valor máximo de pontos para a redução de amostra
27 - stemming = 3; % valor do tampão
28 - maxBurden = 3; % valor do afastamento crítico

```

Figura 22 – Inserção de parâmetros de afastamento crítico e de tamponamento

```

57 % amostra menor para ganhar tempo de processamento
58 - [m n] = size(data);
59 - randomic = randi(m,maxPoints+1000,1);
60
61 % a função randi destina-se a obter uma amostra aleatória dos pontos que
62 % constam na nuvem inicial
63
64 - randomic = unique(randomic);
65 - [m n] = size(randomic);
66
67 - if m>maxPoints
68 -     randomic = randomic(1:maxPoints);
69 - end
70
71 - temp=[];
72
73 - [maxPoints n] = size(randomic);
74
75 - for i=1:maxPoints
76 -     temp=[temp;data(randomic(i),:)];
77 - end
78
79 - data=temp;
80

```

Figura 23 – Exemplo de código para obtenção de uma amostra de pontos reduzida

Após a redução da amostra, é possível obter uma representação gráfica da nuvem de pontos reduzida, observável no lado direito da figura 24.

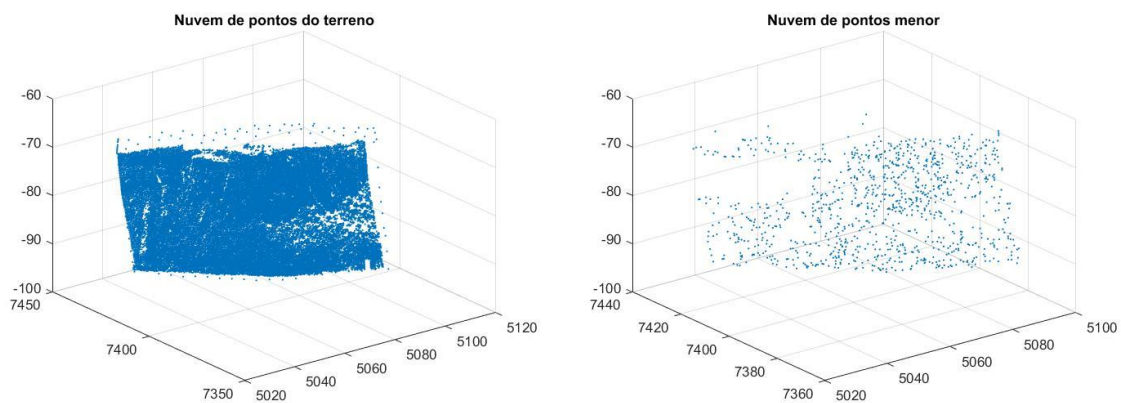


Figura 24 – Comparação entre nuvens de pontos original (à esquerda) e reduzida (à direita)

4.2.2. REPRESENTAÇÃO DOS FUROS E DISCRETIZAÇÃO DOS PONTOS DE CADA FURO

Simultaneamente com a importação dos dados relativos ao terreno, o algoritmo importa um conjunto de dados relativos à primeira linha de furos da bancada (Figura 21), furos esses que se pretendem otimizar em relação à geometria da frente livre. Neste caso, são importados pares de pontos com coordenadas xyz , relativos à posição da crista (*collar*) e do pé (*toe*) de cada furo.

De seguida, e a partir da importação dos dados dos pontos de crista e pé, é possível calcular um vetor unitário \vec{u} :

$$\vec{u} = \frac{toe - collar}{\|toe - collar\|} \quad (\text{Eq. 12})$$

Tomando como exemplo uma bancada cuja altura é de cerca de 15 metros, interessa calcular um número significativo de pontos que sejam capazes de representar o furo de maneira adequada. Uma representação seria considerada fidedigna, se por exemplo, fosse feita à custa de pontos discretizados a cada 30 cm ao longo do furo. Como tal, essa representação teria de ser feita à custa de 50 pontos.

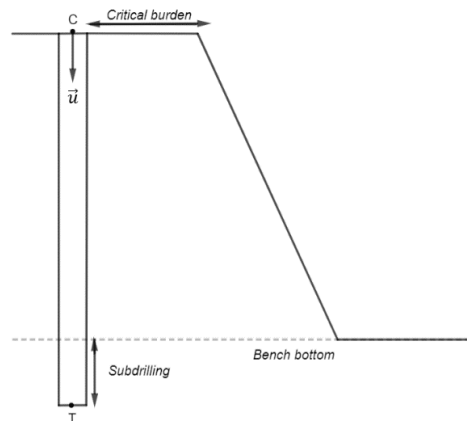


Figura 25 – Representação esquemática do processo de criação de um furo com pontos discretizados

O vetor \vec{u} tem precisamente como objetivo, auxiliar no cálculo da série de pontos discretizados que representam o furo. O vetor, sendo unitário, é multiplicado por um passo (*step*), e pelo índice (i de 1 até n pontos) dos sucessivos pontos a determinar, e quando é somado ao ponto da crista do furo (*collar*), permite determinar as coordenadas de cada ponto do furo:

$$Hole\ point = collar + \vec{u} * step * i \quad (\text{Eq. 13})$$

Sendo o passo obtido através do quociente entre o comprimento desejado para o furo (*hole length*) e o nº de pontos que se deseja discretizar:

$$step = \frac{hole\ length}{number\ of\ points} \quad (Eq. 14)$$

```

104 - for hole =1:m
105 -
106 -     %Gráfico com terreno de amostra reduzida e fila de furos verticais
107 -
108 -     figure.figureNumber;
109 -     figureNumber=figureNumber+1;
110 -
111 -     scatter3(x,y,z,2,'filled');
112 -     hold on;
113 -
114 -     holePoints=[];
115 -     director = toe(hole,:) - collar(hole,:); % vetor diretor do furo
116 -     holeLength = norm(director); % comprimento do furo
117 -     step = holeLength/40; % determinar o passo de discretização
118 -     director=director/holeLength;
119 -     stemmingPosition = collar(hole,:) + director*stemming; %posição do tampão
120 -
121 -
122 -     for i=1:50
123 -
124 -         holePoints=[holePoints;collar(hole,:) + director*step*i];
125 -         % calcula os pontos discretizados ao longo do furo, segundo um
126 -         % passo de discretização, e para os sucessivos índices de 1 até 50
127 -
128 -     end
129 -
130 -     lastToe = holePoints(50,:); % posição do ponto do pé do furo
131 -

```

Figura 26 – Excerto de código para obter a representação dos furos discretizados

O comprimento do furo deverá ser igual ao valor da altura da bancada, somada com o valor da subfuração. Assim, o furo é criado até ao pé da bancada (*bench bottom*), e são acrescentados pontos relativos à subfuração e a uma tolerância. Esta extensão da amostra de pontos é feita de modo a que, os pontos da zona do pé do furo possam ser comparados relativamente a um conjunto de pontos adequados que compõem o pé da bancada. Caso contrário, os pontos da zona do pé do furo poderiam ser comparados relativamente a pontos de zonas superiores ao *bench bottom*, o que faria com que o perfil crítico obtido não correspondesse à realidade, e o processo de otimização não seria devidamente executado.

```

137 % Pontos extra:
138 % são usados para fazer o prolongamento dos pontos do furo relativamente a
139 % frente livre, de modo a que possam ser encontrados pontos-candidatos
140 % adequados aos pontos da zona do pé do furo (preparação para a
141 % obtenção do perfil crítico: adição de 10 pontos
142
143 maxDist=100000;
144 index = 0;
145
146 for j=1:maxPoints
147     M=[lastToe; data(j,:)];
148
149     dist = pdist(M,'euclidean');
150     if dist<maxDist
151         maxDist=dist;
152         index = j;
153     end
154 end
155
156 newDirection = data(index,:) - collar(hole,:);
157
158 newDirection = [newDirection(1) newDirection(2) 0];
159
160 auxLength = 10;
161
162 step = 0.5;
163
164 newDirection=newDirection/auxLength;
165
166 for i=1:10
167     holePoints=[holePoints;lastToe + newDirection*step*i];
168
169 end
170
171

```

Figura 27 – Excerto de código relativo à adição de pontos extra

Assim, é feita a adição de mais 10 pontos de furo, com um passo de 0.5 metros, e são adicionados pontos correspondentes na frente da bancada, com uma distância máxima permitida.

Fazendo a representação tridimensional do código, esta corresponde a uma linha de furos verticais inseridos no terreno (de amostra reduzida), e que serão ajustados mais adiante de acordo com o algoritmo de programação linear.

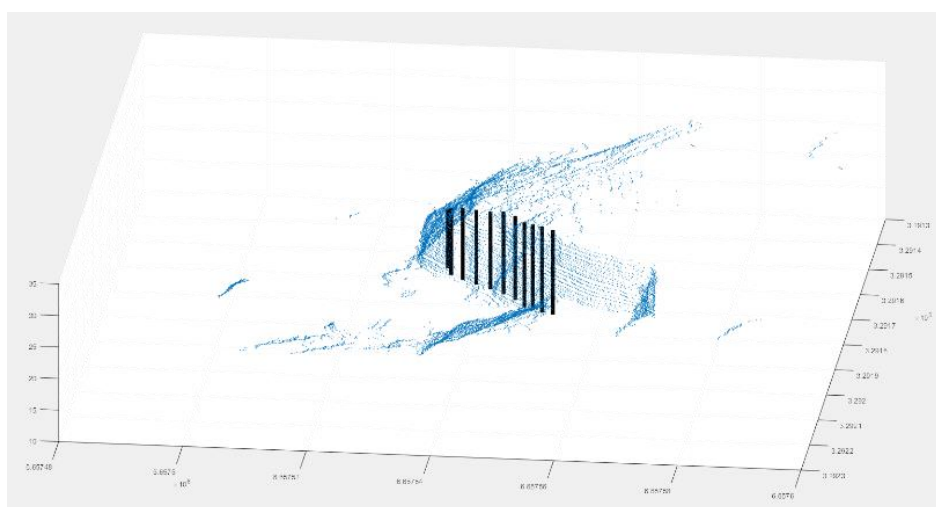


Figura 28 – Representação tridimensional da linha de furos no terreno

4.2.3. OBTENÇÃO DO PERFIL CRÍTICO

Este excerto do algoritmo diz respeito à comparação entre pontos discretizados do furo e pontos da frente livre, sendo obtido um perfil crítico que corresponde à representação em corte, do conjunto de pontos da frente livre que estão mais próximos dos pontos que representam o furo.

Para cada ponto discretizado do furo, é definida uma zona de influência com largura de 2 metros, 1 metro acima do ponto e 1 metro abaixo. Dentro desta zona, estão localizados os “pontos-candidatos”, que correspondem à parcela da nuvem de pontos na mesma cota da zona de influência de cada ponto do furo.

```

175 %afastamento critico
176 [m n] = size(holePoints);
177 criticalBurden=[];
178
179 for i=1:m
180     i
181
182     maxDist=100000;
183     index = 0;
184
185     for j=1:maxPoints
186
187         M=[holePoints(i,:); data(j,:)];
188
189         dist = pdist(M,'euclidean'); %cálculo da distância euclidiana
190
191         % tolerância de +/- 1 metro para cada furo discretizado:
192         % zona de influência
193
194         if dist<maxDist && data(j,3)<holePoints(i,3)+1 && data(j,3)>holePoints(i,3)-1
195             maxDist=dist;
196             index = j;
197
198
199
200         end
201
202     end
203
204     if (index>0)
205         criticalBurden=[criticalBurden;data(index,:)];
206     end
207 end
208

```

Figura 29 – Excerto de código para obtenção do perfil crítico

O algoritmo calcula as distâncias euclidianas entre o ponto discretizado i e os pontos contidos nessa zona de influência, através das coordenadas do ponto do furo (P) e das coordenadas do ponto-candidato (C). A distância euclidiana é definida como sendo a distância entre 2 pontos num espaço euclidiano de coordenadas cartesianas, e que no caso bidimensional é dada pelo teorema de Pitágoras (Boulos, 1987):

$$d_{euclidiana} = \sqrt{(P_x - C_x)^2 + (P_y - C_y)^2} \quad (\text{Eq. 15})$$

Assim, é feito o cálculo da distância entre o ponto P e todos os pontos candidatos, e após encontrar todas as distâncias, é feita uma comparação entre elas. Ao menor valor encontrado para as distâncias, corresponde o ponto candidato mais próximo do furo, logo este ponto pertencerá ao perfil crítico.

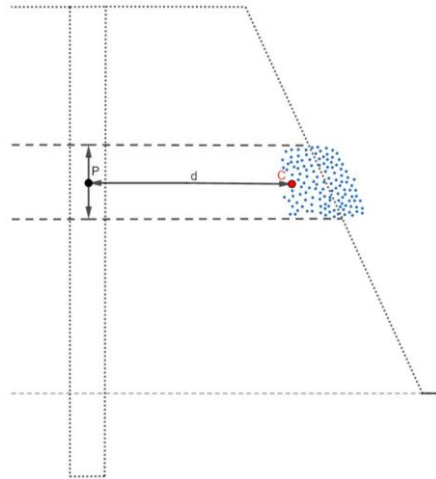


Figura 30 – Esquema representativo de um perfil de bancada com nuvem de pontos candidatos

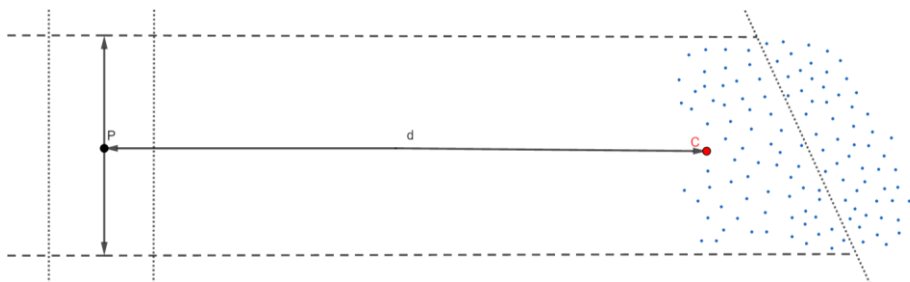


Figura 31 – Pormenor da zona de influência do ponto P do furo, e distância entre um ponto candidato C

Este cálculo é feito de $i=1$ até todos os n pontos ao longo do furo, isto é, o cálculo das distâncias é sequencial e avança desde o primeiro até ao último ponto discretizado que compõe o furo, e o resultado obtido é um perfil crítico cujos pontos estão em quantidade igual ao número de pontos discretizados do furo. A obtenção do perfil crítico é feita para n furos, e cada furo terá um perfil crítico próprio e distinto dos outros furos, logo serão obtidos igualmente n perfis críticos.

A definição dos perfis críticos é de extrema importância para os resultados da otimização, uma vez que o ajuste da inclinação do furo dependerá sempre das características geométricas do perfil crítico ao qual o furo deve ser ajustado.

4.2.4. DETEÇÃO E EXCLUSÃO DE *OUTLIERS*

Num dado conjunto de observações, um *outlier* define-se como um objeto (ou conjunto de objetos) que se encontra desviado ou isolado das restantes observações (Aziz, 2012).

A ocorrência de *outliers* num conjunto de dados pode levar a leituras erradas sobre o fenómeno em estudo. Há casos onde a ocorrência de *outliers* é demasiado evidente, com um ou mais pontos a assumirem valores bastante anómalos relativamente ao resto da amostra, enquanto noutros casos é necessário recorrer a ferramentas capazes de realizar a sua deteção (Miranda, 2016).

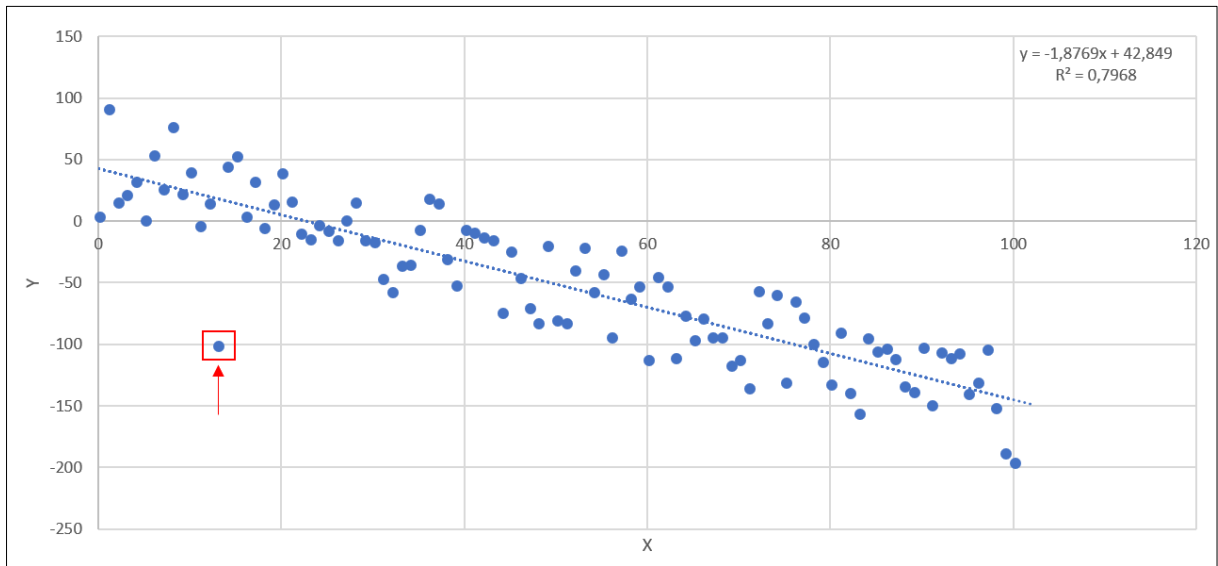


Figura 32 – Gráfico de uma reta de regressão correspondente a um conjunto de dados com um *outlier*

Num dado conjunto de pontos de uma amostra, se esta for correspondente a um espaço bidimensional, é possível obter um hiperplano (neste caso, uma reta) que ajusta os pontos através de regressão linear por mínimos quadrados (Gujarati, 1978). Observando a Figura 32, num conjunto de dados ligeiramente dispersos ajustados a uma reta, é encontrado um ponto correspondente a um *outlier*, que foi detetado com recurso a um dos métodos de deteção de *outliers*, que consiste no seguinte procedimento (Gama, 2010):

- Calcular o 1º e 3º quartis do conjunto de dados;
- Calcular o intervalo interquartil:

$$(IQR = Q1 - Q3) \quad (\text{Eq. 16})$$

- Classificar um dado ponto x como *outlier* moderado caso:

$$Q3 + 1.5 * IQR < x < Q1 - 1.5 * IQR \quad (\text{Eq. 17})$$

No anexo A.3 está presente uma função MATLAB denominada *EraseOutliers.m* que funciona como uma ferramenta de detecção e subtração de *outliers* dos pontos correspondentes ao perfil crítico.

O algoritmo devolve cada perfil crítico como uma variável denominada *criticalBurden*, cujo conjunto de pontos pode conter alguns pontos desviados do resto do conjunto. Assim, e dada a importância da definição do perfil crítico, torna-se essencial o recurso a uma ferramenta que seja capaz de filtrar esses pontos e que devolva um conjunto de pontos que seja adequado à operação de otimização.

A função *EraseOutliers.m* recorre a uma outra função (anexo A.2.) denominada *projectPointinLine.m* cujo objetivo é efetuar o cálculo da projeção vetorial dos pontos do perfil crítico na reta do furo.

4.2.5.1. Projeção vetorial de um ponto numa reta – função *projectPointinLine.m*

Tomando como exemplo um perfil de bancada como o da Figura 33, em que os pontos C e T correspondem, respetivamente, ao *collar* (crista) e ao *toe* (pé) do furo, e em que o ponto B representa um ponto pertencente ao perfil crítico da bancada, é possível definir dois vetores:

$$\overrightarrow{CT} = T - C \quad (\text{Eq. 18})$$

e

$$\overrightarrow{CB} = B - C \quad (\text{Eq. 19})$$

Em que \overrightarrow{CT} representa o vetor diretor da reta correspondente ao furo, e \overrightarrow{CB} corresponde ao vetor que liga o ponto da crista ao ponto do perfil crítico.

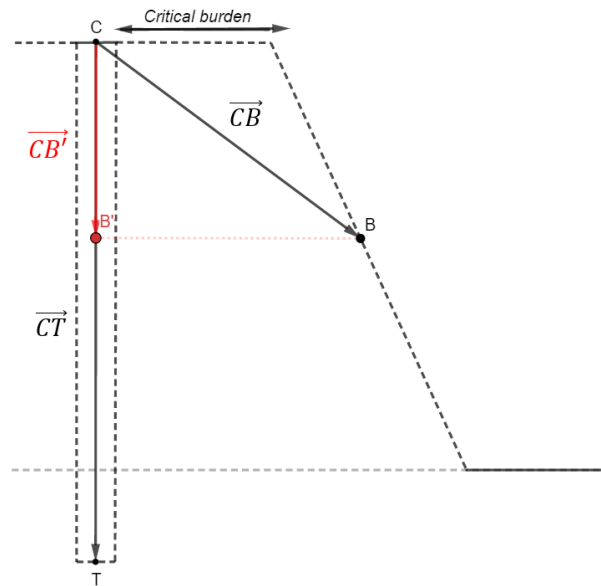


Figura 33 – Projeção de um ponto do perfil crítico na reta do furo

O objetivo é obter a projeção do ponto B (B'), na reta que representa o furo, e para tal recorre-se à projeção vetorial de \overrightarrow{CB} em \overrightarrow{CT} (Boulos, 1987):

$$\overrightarrow{CB'} = \text{proj}_{\overrightarrow{CT}} \overrightarrow{CB} = \frac{\overrightarrow{CB} \cdot \overrightarrow{CT}}{\|\overrightarrow{CT}\|^2} * \overrightarrow{CT} \quad (\text{Eq. 20})$$

em que a primeira parcela $\frac{\overrightarrow{CB} \cdot \overrightarrow{CT}}{\|\overrightarrow{CT}\|^2}$ corresponde à magnitude do vetor $\overrightarrow{CB'}$ e a segunda parcela $\frac{\overrightarrow{CT}}{\|\overrightarrow{CT}\|}$ dará a direção desse vetor. A soma do vetor $\overrightarrow{CB'}$ com o ponto C, permite obter a posição do ponto B' na reta do furo.

Este cálculo é, naturalmente, executado para todos os pontos pertencentes ao perfil crítico.

```

1  function projection = projectPointInLine(l1,l2,p)
2
3  % Esta função destina-se ao cálculo da projeção dos pontos da frente livre
4  % na reta que representa o furo
5
6
7  [m n] = size(p);
8
9  projection=[];
10
11  for i=1:m
12
13      % vetor u, da crista ao ponto do perfil
14      u = p(i,:)-l1;
15
16      % vetor v, vetor diretor da reta que representa o furo
17      v=l2-l1;
18      director = l2-l1;
19      director = director/norm(director);
20
21      % Pontos projetados
22      projection=[projection; l1 + director*dot(u,v)/norm(v)];
23
24  end
25
26
27 end

```

Figura 34 – Função *projectPointInLine.m*

4.2.5.2. Detecção e exclusão dos *outliers* com recurso à função *EraseOutliers.m*

Tendo obtido as projeções de todos os pontos da frente livre ao longo da reta do furo através da função *projectPointInLine.m*, a função *eraseOutliers.m* calcula a distância euclidiana tridimensional entre cada ponto do perfil crítico e cada ponto projetado na reta do furo:

$$d_{euclidiana} = \sqrt{(P_x - C_x)^2 + (P_y - C_y)^2 + (P_z - C_z)^2} \quad (\text{Eq. 21})$$

Uma vez calculadas todas as distâncias para os n pares de pontos, o vetor correspondente a esses valores é ordenado e é feito o cálculo do 1º e 3º quartis da amostra em causa, sendo finalmente obtido o intervalo inter-quartil (Eq. 16).

Obtendo o intervalo inter-quartil da distância dos pontos do perfil crítico aos pontos projetados no furo, é possível classificar os pontos como *outliers* ou não, através da (Eq. 17) vista anteriormente. A função *eraseOutliers.m* devolverá assim um vetor que corresponde ao output do perfil crítico já sem os pontos que tenham sido identificados como *outliers* anteriormente.

```

1 function result = eraseOutliers( criticalBurden, collar, toe )
2
3 % Esta função destina-se à deteção e exclusão de outliers presentes no
4 % perfil crítico
5
6 projection = projectPointInLine(collar,toe,criticalBurden);
7
8 distance = [];
9
10 [m n] = size(projection);
11
12 %cálculo das distâncias entre perfil crítico e projeção dos pontos
13
14 for i=1:m
15     distance=[distance; ((criticalBurden(i,1)-projection(i,1))^2 + ...
16         (criticalBurden(i,2)-projection(i,2))^2 + (criticalBurden(i,3)-projection(i,3))^2)^0.5];
17 end
18
19 % ordenar dados
20
21 orderedDistance = sort(distance);
22
23 % cálculo inter-quartil
24
25 q25 = quantile(orderedDistance, 0.25);
26 q75 = quantile(orderedDistance, 0.75);
27
28 difInterQuantile = q75-q25;
29
30 result=[];
31
32 for i=1:m
33
34     if (distance(i)>q25-1.5*difInterQuantile && distance(i)<q75+1.5*difInterQuantile)
35
36         result = [result;criticalBurden(i,:)];
37         % output do perfil crítico sem outliers
38     end
39 end
40
41 end
42

```

Figura 35 – Função *eraseOutliers.m*

4.2.5. CÁLCULO DA DIREÇÃO DE DISPERSÃO DO PERFIL CRÍTICO – OBTENÇÃO DE NOVA BASE

Após o cálculo do perfil crítico e da filtragem dos *outliers*, seria de esperar que o próximo passo fosse a programação linear do problema e a sua resolução. No entanto, é necessário proceder a alguns ajustes essenciais para que o algoritmo encontre uma solução ótima e viável.

Um desses ajustes passa pela manipulação do perfil crítico, através de uma mudança de base e de uma translação, de modo a que o furo, representado por uma reta de equação $y=ax+b$, esteja em condições de ser ajustado de maneira correta e ótima ao perfil crítico.

4.2.5.1. Matriz de covariância

Para realizar esta manipulação, é necessário definir alguns passos prévios, sendo o primeiro desses passos o cálculo da matriz de covariância dos dados.

Em Estatística, a covariância define-se como uma medida do grau de correlação entre duas variáveis. Considerando duas variáveis aleatórias X e Y , e μ_x e μ_y os valores médios de X e Y , a covariância entre X e Y é dada por (Murteira, et al., 2008):

$$\sigma_{xy} = E[(X - \mu_x)(Y - \mu_y)] = E(XY) - E(X)E(Y) \quad (\text{Eq. 22})$$

A correlação entre as duas variáveis é nula se $\sigma_{xy}=0$, positiva ($\sigma_{xy}>0$) se Y crescer com o aumento de X , e negativa ($\sigma_{xy}<0$) se Y decrescer com o aumento de X . Quando $X=Y$, a covariância σ_{xy} pode ser dada por $\sigma_{xx} = E(X^2) - E(X)^2$, o que é equivalente à variância σ_x^2 (Weisstein).

Uma matriz de covariância é uma matriz quadrada $n \times n$, em que n é a dimensão do espaço em que se está a trabalhar, logo, uma matriz relativa a dados tridimensionais terá dimensão 3×3 . Na diagonal da matriz de covariância estão dispostos os elementos relativos às variâncias e fora da diagonal estão os elementos relativos às covariâncias, sendo a matriz simétrica. A matriz pode então ser formulada da seguinte forma, para um dado vetor v :

$$\text{Cov}(v) = \begin{bmatrix} \sigma_x^2 & \sigma_{xy} & \sigma_{xz} \\ \sigma_{xy} & \sigma_y^2 & \sigma_{yz} \\ \sigma_{xz} & \sigma_{yz} & \sigma_z^2 \end{bmatrix} \quad (\text{Eq. 23})$$

Assumindo que o vetor é o do perfil crítico (vetor *criticalBurden*), é possível calcular a matriz de covariância para o perfil crítico obtido anteriormente. Esta matriz define a dispersão (através das variâncias) e a orientação espacial (através das covariâncias) dos pontos que estão associados ao perfil crítico.

4.2.5.2. Valores e vetores próprios

Dada uma matriz quadrada A de ordem n , como é o caso da matriz de covariância, é possível achar um escalar λ (valor próprio de A), se existir um vetor próprio v não-nulo para o qual é válida a igualdade: $Av = \lambda v$. Atendendo à dimensão $n \times n$ da matriz, é possível obter n valores próprios associados a n vetores próprios (Lipschutz, 1991).

No caso da matriz de covariância, de dimensão 3×3 , estão associados 3 valores próprios e 3 vetores próprios. Na figura Figura 36 está representado um conjunto de pontos com coordenadas (x,y) , e dois vetores u e v que correspondem aos vetores próprios associados à matriz de covariância dos dados.

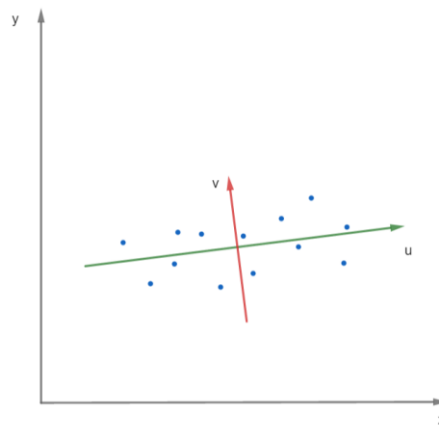


Figura 36 – Dispersão de um conjunto de pontos e vetores próprios associados

O vetor u aponta a direção de maior dispersão dos dados, e como tal será o vetor próprio associado ao maior valor próprio da matriz de covariância, enquanto que o vetor v , ortogonal a u , aponta na direção de segunda maior dispersão dos dados. Estando perante um caso tridimensional (Figura 37), seria obtido ainda um terceiro vetor ortogonal a ambos, que apontaria na terceira maior direção de dispersão dos dados contidos dentro da elipsoide.

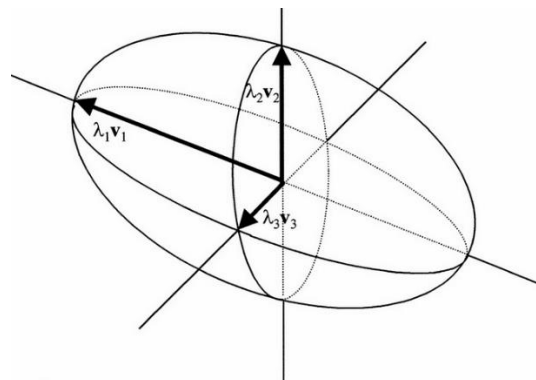


Figura 37 – Vetores próprios direcionais para o caso tridimensional
(<http://radiology.rsna.org/content/217/3/897/F1.expansion.html>)

Considerando o conjunto de dados tridimensionais relativos ao perfil crítico, é possível calcular a matriz de covariância e os valores próprios e vetores próprios associados a essa matriz. Encontrando o maior valor próprio e o respetivo vetor próprio que lhe está associado, obtém-se o vetor próprio que indica a direção que os pontos do perfil crítico tomam, pois aponta na direção de maior dispersão dos dados, à semelhança do vetor u das figuras Figura 38 e Figura 40.

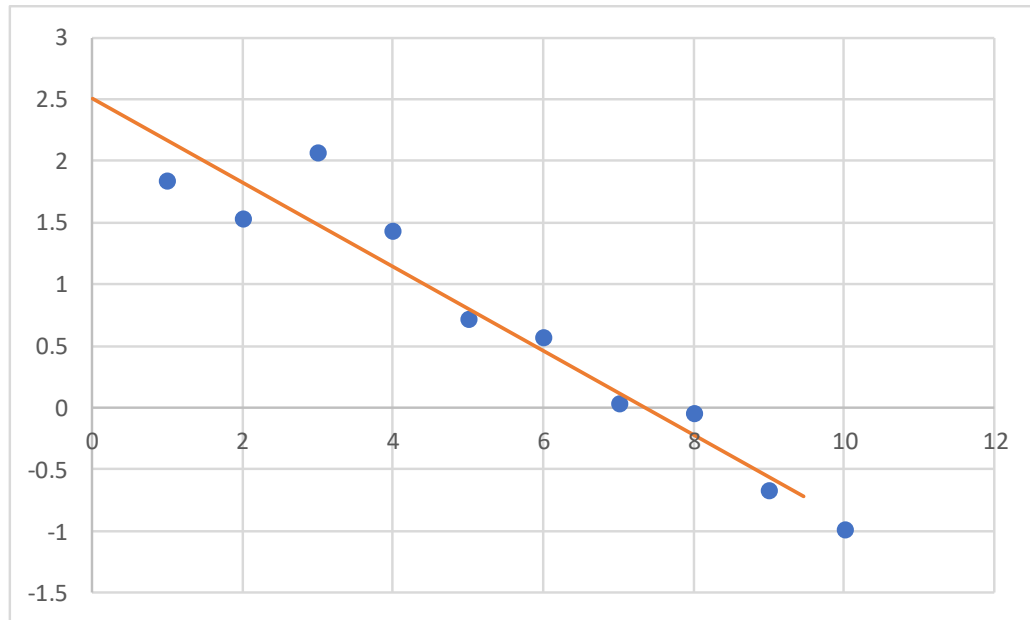


Figura 38 – Vetor próprio que indica a direção de dispersão de um conjunto de dados (representação bidimensional)

Assim, para este problema, é possível definir o vetor que indica a direção de maior dispersão dos dados, como um eixo que aponta no sentido e direção do perfil crítico, e como tal, pode utilizar-se este vetor como se se tratasse de um eixo de referência para uma transformação linear de mudança de base. Esta transformação terá como objetivo a definição de novos eixos coordenados nos quais o perfil estará representado, e permitirá trabalhar o perfil crítico de modo a que seja possível o ajuste de uma reta representativa de um furo.

```

249 % Covariância
250
251 covar = cov([criticalBurdenVariance(:,1),...
252             criticalBurdenVariance(:,2),criticalBurdenVariance(:,3)]);
253
254 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
255
256 %Cálculo dos valores e vetores próprios
257
258 [AVe AVa] = eig(covar);
259
260 Vector = [];
261
262 % Encontrar maior valor próprio e vetor próprio associado
263
264 if AVa(3,3)>AVa(2,2) && AVa(3,3)>AVa(1,1)
265
266     Vector=AVe(:,3);
267
268 else
269
270     if AVa(2,2)>AVa(1,1)
271         Vector=AVe(:,2);
272     else
273         Vector=AVe(:,1);
274     end
275
276 end
277

```

Figura 39 – Cálculo da matriz de covariância e do vetor próprio associado ao maior valor próprio

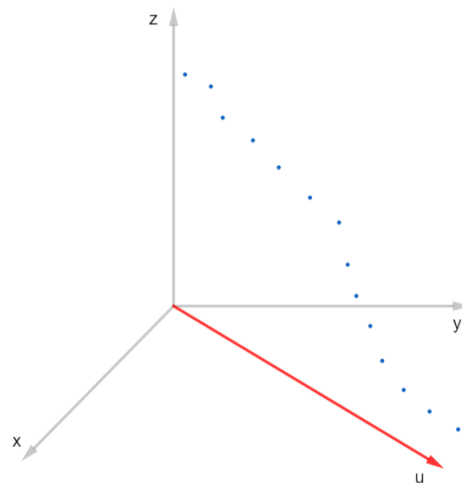


Figura 40 – Vetor próprio que indica a direção de dispersão do perfil crítico (representação tridimensional)

4.2.6. MUDANÇA DE BASE

O objetivo da mudança de base é obter uma representação do perfil crítico que seja mais conveniente para a resolução do problema. Como será visto mais adiante, a modelação do furo está intrinsecamente dependente da mudança de base e de uma representação modificada do perfil crítico.

4.2.6.1. Base

Uma base de um espaço vetorial E , define-se como um conjunto de vetores $\{e_1, e_2, \dots, e_n\}$ linearmente independentes entre si, e cuja combinação linear é capaz de gerar o espaço vetorial (Boulos, 1987).

Uma base canónica é um conjunto de vetores unitários que apontam na direção dos eixos coordenados de um espaço cartesiano. Para um espaço tridimensional \mathbb{R}^3 , a base canónica é formada pelos vetores $\{e_x, e_y, e_z\} = \{(1,0,0), (0,1,0), (0,0,1)\}$ (Ryan, 1986).

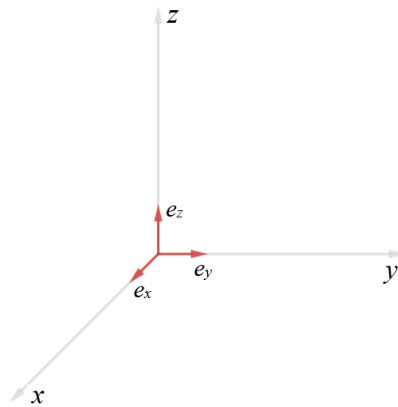


Figura 41 – Representação gráfica de uma base canónica em \mathbb{R}^3

Todos os vetores que são trabalhados e obtidos ao longo do algoritmo, estão representados na base canónica, incluindo o vetor *criticalBurden*, que representa o perfil crítico. Para que a sua representação seja adequada ao modelo, é necessário recorrer a uma mudança de base em que um dos vetores que formam a nova base seja o vetor que aponta na direção do perfil crítico.

4.2.6.2. Nova base

Achando o vetor u (Figura 40), que formará um novo sistema de eixos, é possível formar uma nova base de coordenadas. A nova base $\{x', y', z'\}$ será composta pelos seguintes vetores:

- Componente x' – dada pela projeção do vetor u na origem, que aponta na direção de dispersão dos pontos do perfil crítico;
- Componente z' – dada pelo vetor diretor do eixo zz , logo terá coordenadas $(0,0,1)$;
- Componente y' – ortogonal a x e z , é dada pelo produto vetorial $u \times z$.

```

278 % Base nova, formada pelo vetor próprio associado ao maior valor
279
280 - baseX = [Vector(1) Vector(2) 0];
281 - baseX=baseX/norm(baseX);
282 - baseZ=[0 0 1];
283 - baseY = cross(baseX,baseZ);
284 - baseY=baseY/norm(baseY);
285
286 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

Figura 42 – Formação de base nova

As componentes são ainda divididas pelos comprimentos dos respectivos vetores, de modo a serem expressas como vetores unitários.

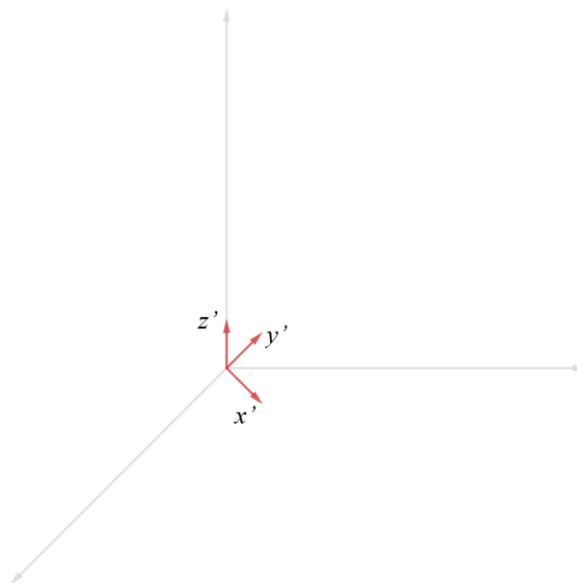


Figura 43 – Representação gráfica dos novos eixos coordenados

4.2.6.3. Conceito de mudança de base segundo Boulos (1987):

Para poder ser realizada a mudança de base, é necessário conhecer a relação entre ambas as bases. Considerando as bases $E = (e_1, e_2, e_3)$ e $F = (f_1, f_2, f_3)$, é possível expressar os elementos de F em relação à base E , da seguinte forma (Boulos, 1987):

$$f_1 = a_{11}e_1 + a_{21}e_2 + a_{31}e_3 \quad (\text{Eq. 24})$$

$$f_2 = a_{12}e_1 + a_{22}e_2 + a_{32}e_3 \quad (\text{Eq. 25})$$

$$f_3 = a_{13}e_1 + a_{23}e_2 + a_{33}e_3 \quad (\text{Eq. 26})$$

Assim, tomando como exemplo um vetor v , em ambas as bases E e F :

$$v_E = xe_1 + ye_2 + ze_3 \quad (\text{Eq. 27})$$

$$v_F = x'f_1 + y'f_2 + z'f_3 \quad (\text{Eq. 28})$$

É possível estabelecer uma relação de coordenadas entre (x, y, z) na base E e (x', y', z') na base F , substituindo as componentes da base F (f_1, f_2, f_3) em v_F :

$$v_F = x'(a_{11}e_1 + a_{21}e_2 + a_{31}e_3) + y'(a_{12}e_1 + a_{22}e_2 + a_{32}e_3) + z'(a_{13}e_1 + a_{23}e_2 + a_{33}e_3) \quad (\text{Eq. 29})$$

Fazendo a comparação com v_E , e sabendo que um vetor pode ser obtido através da combinação linear de componentes de uma base, tem-se:

$$x = a_{11}x' + a_{21}y' + a_{31}z' \quad (\text{Eq. 30})$$

$$y = a_{12}x' + a_{22}y' + a_{32}z' \quad (\text{Eq. 31})$$

$$z = a_{13}x' + a_{23}y' + a_{33}z' \quad (\text{Eq. 32})$$

O que equivale a escrever a seguinte equação matricial:

$$v_E = M * v_F \Leftrightarrow \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} * \begin{bmatrix} x' \\ y' \\ z' \end{bmatrix}, \quad (\text{Eq. 33})$$

onde M é a matriz de mudança de base $E \xrightarrow{M} F$.

Como o determinante da matriz M é diferente de zero, existe a matriz inversa M^{-1} , que permite ser possível obter a relação:

$$v_F = M^{-1} * v_E \quad (\text{Eq. 34})$$

4.2.6.3. Mudança de base recorrendo à função *getPointsNewBase.m*:

A função presente no anexo A.4. permite fazer a operação de mudança de base, utilizando os pressupostos observados anteriormente. É definida uma matriz de mudança de base, formada pelos coeficientes das componentes da nova base. Esta matriz é invertida, uma vez que são conhecidos os valores do perfil crítico na base canónica, e pretende-se obter os valores do perfil crítico na nova base (Eq. 34).

```

1 function points = getPointsNewBase(baseX,baseY,baseZ, points, collar, toe)
2
3 % Esta função destina-se ao cálculo dos pontos que formam a nova base
4
5 M=[baseX(1) baseY(1) baseZ(1); baseX(2) baseY(2) baseZ(2); baseX(3) baseY(3) baseZ(3)];
6
7 % matriz de mudança de base
8
9 M=M^(-1);
10
11 [m n] = size(points);
12
13 for i=1:m
14
15     %projeção dos pontos
16
17     projection = projectPointInLine(collar,toe,points(i,:));
18
19     % distância euclidiana entre furo e perfil
20
21     dist = pdist([points(i,:); projection],'euclidean');
22
23     % mudança de base
24
25     points(i,:) = (M*points(i,:))';
26
27     % coordenada y = distância entre furo e pontos
28
29     points(i,2) = dist;
30
31 end
32
33 end

```

Figura 44 – Função *getPointNewBase.m*

Esta função faz também a projeção dos pontos e calcula as distâncias entre os pontos na nova base e a sua projeção. Esta distância corresponderá aos valores em y dos pontos na nova base. A figura Figura 45 é representativa daquilo que se pretende com esta operação. É feita uma rotação de 90° do perfil crítico e torna-se evidente que esta manipulação do perfil crítico servirá para que a reta do furo, inicialmente vertical, seja coincidente com o eixo das abcissas, sendo que após a etapa de programação linear passará a ser uma reta inclinada – com inclinação otimizada.

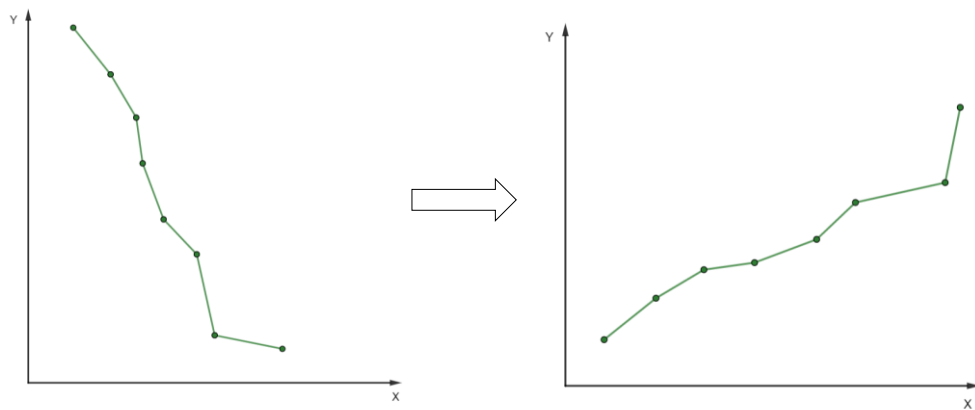


Figura 45 – Perfil crítico na nova base e rotação de 90° (representação bidimensional)

4.2.7. TRANSLAÇÃO E PERFIL CRÍTICO FINAL

Ainda antes de proceder à programação linear, é necessário fazer um pequeno ajuste, uma vez que o perfil crítico está deslocado em relação aos eixos coordenados. Para tal é necessário calcular os pontos mínimos $\{mín(x), mín(y), mín(z)\}$, e subtrair estes mínimos aos sucessivos pontos que compõem o perfil crítico. Isto fará com que seja feita uma translação do perfil crítico para a origem (Figura 46).

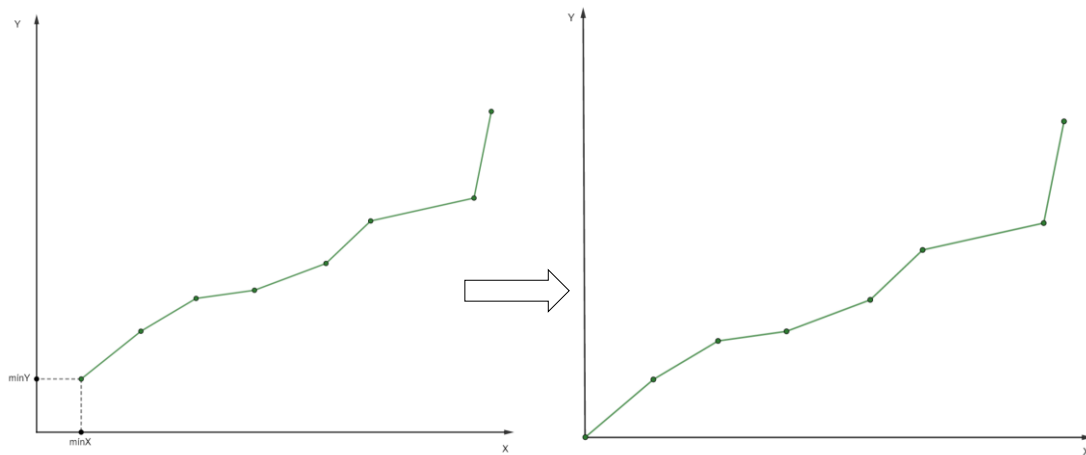


Figura 46 - Translação do perfil crítico (representação bidimensional)

4.3. RESOLUÇÃO DO PROBLEMA UTILIZANDO PROGRAMAÇÃO LINEAR EM MATLAB

Tendo os dados todos preparados, nomeadamente no que ao perfil crítico diz respeito, procede-se à execução do algoritmo de otimização. Para este efeito, é usada a função *linprog*, uma das ferramentas de otimização do MATLAB, previamente explanada no capítulo 3.

4.3.1. MODELO MATEMÁTICO DE UM FURO

Um furo de bancada, no espaço tridimensional, é uma cavidade cilíndrica com um determinado diâmetro e comprimento. Um furo ideal, é uma cavidade cilíndrica perfeita, cujo perfil é semelhante a uma reta. Na realidade, ocorrem desvios na esmagadora maioria das vezes, que fazem com que o perfil do furo deixe de ser uma reta, e passe a ter um formato irregular e “curvo”.

Para definir uma reta, são apenas necessários dois pontos conhecidos. A partir desses pontos é possível determinar o vetor diretor da reta, e a partir do vetor diretor é conhecida a equação vetorial da reta $R=A+\lambda v$, onde A é um ponto da reta, λ um escalar real qualquer, e v o vetor diretor da reta (Boulos, 1987).

Matematicamente, e desprezando os desvios que ocorrem frequentemente por erro do perfurista ou mau funcionamento, um furo poderá ser considerado, num espaço cartesiano bidimensional \mathbb{R}^2 , como uma aproximação a uma reta real de equação reduzida $y=ax+b$, sendo:

- y e x , as coordenadas cartesianas do furo;
- a , o declive da reta;
- b , a ordenada na origem.

Os parâmetros a e b são desconhecidos *a priori*, uma vez que a equação da reta só ficará definida após encontrar a solução ótima para o problema. Estes parâmetros constituem assim duas variáveis de decisão para o problema.

4.3.2. CONSTRUÇÃO DO MODELO DO FURO

Uma vez que a inclinação do furo é afetada pelo comportamento geométrico da bancada, é necessário trabalhar com os dados do perfil crítico, que deverão ter sido previamente tratados pelo algoritmo MATLAB visto anteriormente.

Como ponto de partida, considera-se um furo de equação $y=ax+b$, perfeitamente vertical (representado na Figura 47 pelo eixo xx), e discretizado em X_n pontos, que se encontra a uma distância B (de 1 até n) dos Y_n pontos correspondentes ao perfil crítico.

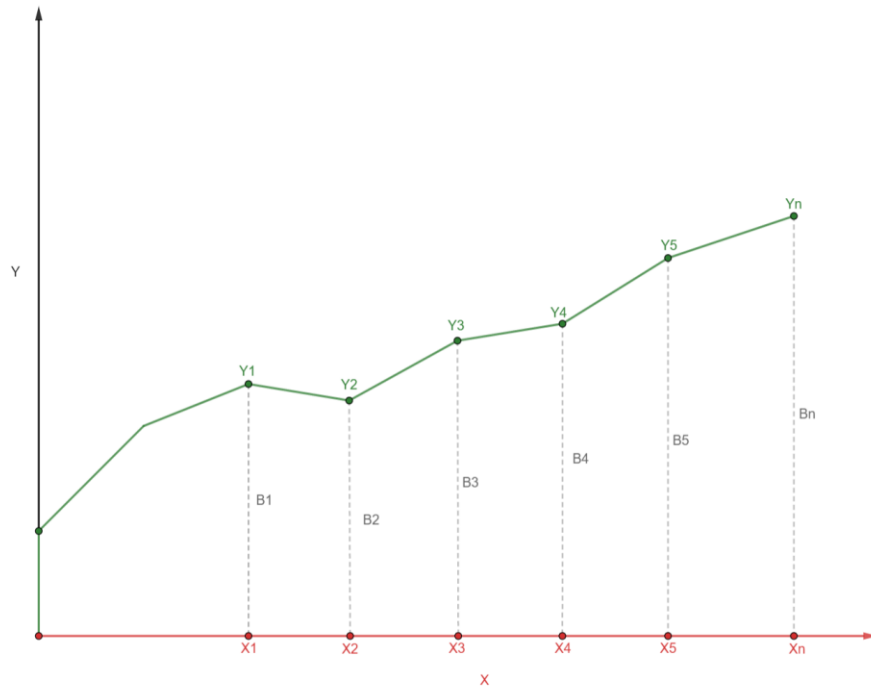


Figura 47 – Afastamento entre pontos de frente livre (Y_n) e pontos discretizados de um furo vertical (X_n)

A distância B , que é precisamente o afastamento crítico, é dada pela diferença entre os pontos do perfil crítico e os pontos do furo:

$$B_i = y_{frente} - y_{furo} \quad (\text{Eq. 35})$$

Contudo, é sabido que o valor de y_{furo} é dado pela equação do modelo do furo:

$$y_{furo} = a + b.X \quad (\text{Eq. 36})$$

O que, tomando como exemplo a figuraFigura 47, se traduz na seguinte relação para o valor do afastamento:

$$B_1 = Y_1 - (a + b.X_1) \quad (\text{Eq. 37})$$

$$B_2 = Y_2 - (a + b.X_2) \quad (\text{Eq. 38})$$

(...)

$$B_n = Y_n - (a + b.X_n) \quad (\text{Eq. 39})$$

Os sucessivos afastamentos B_i têm diferenças entre si, que são representadas por um parâmetro denominado ΔB_i . Esta diferença entre afastamentos sucessivos deverá ser o menor possível, de modo a uniformizar o afastamento ao longo do furo, e por consequência, obter uma melhor distribuição de carga ao longo do furo.

Os vários valores ΔB_i podem ser calculados do seguinte modo, para $i=1$ até $i=n-1$, sendo n o número de pontos utilizados para discretizar o furo:

$$\Delta B_1 = B_1 - B_2 \quad (\text{Eq. 40})$$

$$\Delta B_2 = B_2 - B_3 \quad (\text{Eq. 41})$$

(...)

$$\Delta B_{n-1} = B_{n-1} - B_n \quad (\text{Eq. 42})$$

Este parâmetro servirá para controlar a inclinação do furo, quando associada a uma restrição com valor-limite, como será visto de seguida.

4.3.3. VARIÁVEIS DE DECISÃO

Como já foi referido anteriormente, os parâmetros relativos ao declive e à ordenada na origem da reta representativa do furo, respetivamente a e b , são variáveis de decisão.

Outra das variáveis de decisão é o parâmetro θ_i . Este parâmetro corresponde a um conjunto de valores-limite que todos os ΔB_i poderão assumir após ser realizada a otimização pelo algoritmo, isto é, as diferenças sucessivas de afastamento entre pontos (em módulo) não poderão exceder os valores que o algoritmo indica ao encontrar uma solução ótima. Isto constitui uma restrição para o problema, e o problema terá tantas restrições quantos os pontos que estão discretizados e formam o perfil crítico.

$$|\Delta B_i| \leq \theta_i \rightarrow -\theta_i \leq \Delta B_i \leq \theta_i, \text{ para } i=1 \text{ até } n-1 \quad (\text{Eq. 43})$$

4.3.4. FUNÇÃO OBJETIVO

Neste problema, a função objetivo é a minimização da soma dos valores-limite, $\sum \theta_i$, com a soma dos valores do afastamento, $\sum B_i$, ou seja:

$$\text{Função objetivo} = \min\{\sum \theta_i + \sum B_i\}, \text{ para } i=1 \text{ até } n-1 \quad (\text{Eq. 44})$$

O objetivo deste algoritmo de otimização é no fundo, encontrar um mínimo global ótimo para esta soma, alterando as variáveis de decisão referidas anteriormente, de modo a que os sucessivos afastamentos entre pontos da frente livre e os pontos discretizados do furo sejam o mais uniformes possível. Isto poderia ser obtido à custa da minimização da parcela $\sum \theta_i$, referente aos valores-limite. No entanto, a solução não garantiria que o furo ficasse com um afastamento apropriado em relação à frente, podendo ficar demasiado afastado, e como tal, torna-se necessário garantir que isso não aconteça, minimizando também a parcela $\sum B_i$.

4.3.5. RESTRIÇÕES

Na modelação em MATLAB foram consideradas as seguintes restrições:

- $B_{non-Stemming} \geq B_{ideal}$, restringe todos os pontos no intervalo $y =]final\ do\ tampão,\ pé\ do\ furo]$, de modo a que o seu afastamento em relação à face livre seja igual ou superior ao afastamento crítico definido;
- $-\theta_i \leq \Delta B_i \leq \theta_i$, que se desdobra em duas restrições:
 - $\Delta B_i \leq \theta_i$, restringe todas as diferenças sucessivas de afastamento a serem iguais ou inferiores aos valores-limite;
 - $\Delta B_i \geq -\theta_i$, restringe todas as diferenças sucessivas de afastamento a serem iguais ou superiores ao simétrico dos valores-limite;
- $\theta_i \geq 0$, restringe todos os valores θ_i a valores não-negativos

4.3.6. RESOLUÇÃO COM A FUNÇÃO LINPROG

4.3.6.1. Otimização a partir do tampão

Em primeiro lugar, é necessário referir que na formulação deste algoritmo optou-se por desprezar os pontos correspondentes à zona do tampão, e proceder à otimização a partir dos pontos discretizados imediatamente a seguir, uma vez que a zona do tampão não tem carga explosiva. Isto é, foram desprezados os pontos no intervalo $y = [crista,\ final\ do\ tampão]$, e foram apenas sujeitos a otimização os pontos correspondentes ao intervalo $y =]final\ do\ tampão,\ pé\ do\ furo]$ ².

² Conforme foi sugerido pelo Professor Doutor Alexandre Leite (2018), aquando da idealização desta Dissertação. Este pormenor torna o processo de otimização mais leve, sobretudo em ambiente limitado como é o caso do Excel, como será visto no ponto 4.4.

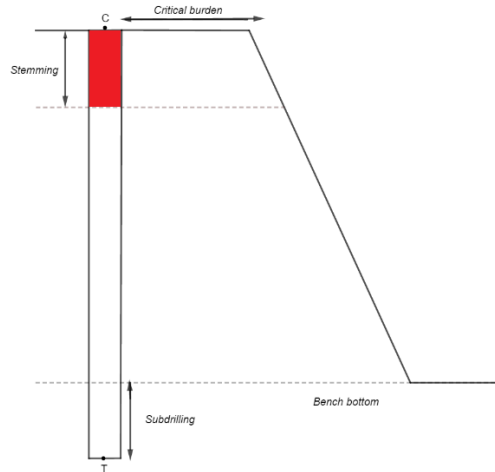


Figura 48 - Representação esquemática do furo com a zona de tampão assinalada a vermelho

Assim, é inserido um vetor nulo *dataSolver* que irá assumir os valores do vetor *criticalBurden*, apenas a partir dos pontos fora do tampão. Importa ainda recordar que o valor do tampão (variável *stemming*) terá sido definido anteriormente, no início do algoritmo (ponto 4.2.1.), e pode ser alterado consoante o pretendido pelo operador.

```

315 - dataSolver=[];
316 -
317 - [m n] = size(criticalBurden);
318 -
319 - % fazer a otimização apenas a partir do tampão
320 -
321 - for i=1:m
322 -
323 -     if (criticalBurden(i,1)>stemming)
324 -
325 -         dataSolver=[dataSolver;criticalBurden(i,:)];
326 -     end
327 -
328 - end
329 -

```

Figura 49 – Etapa inicial do processo de otimização, considerando apenas os pontos do furo abaixo da zona de tampão

4.3.6.2. Formulação da matriz *A* e do vetor *b*, variáveis de entrada na função *linprog*

Como foi visto no capítulo 3, a função *linprog* tem como objetivo encontrar um mínimo global ótimo para o problema, e trabalha da seguinte forma:

$$\min_x f^T x \mid A \cdot x \leq b \quad (\text{Eq. 45})$$

Utilizando os pressupostos observados ao longo deste subcapítulo 4.3., tem-se como variável de decisão, os valores-limite θ_i :

$$-\theta_i \leq \Delta B_i \leq \theta_i \quad (\text{Eq. 46})$$

Contudo, como o problema deve ser formulado segundo a minimização observada na inequação (Eq. 45), a inequação (Eq. 46) terá de ser desdobrada da seguinte forma:

$$\Delta B_i \leq \theta_i \quad (\text{Eq. 47})$$

e

$$-\Delta B_i \leq \theta_i \quad (\text{Eq. 48})$$

Desenvolvendo as inequações, obtém-se (para $i=1$ até $n-1$):

$$B_i - B_{i+1} \leq \theta_i \quad (\text{Eq. 49})$$

e

$$-B_i + B_{i+1} \leq \theta_i \quad (\text{Eq. 50})$$

Como foi visto anteriormente, em 4.3.2.: $B_{i+1} = Y_{i+1} - (a + b.X_{i+1})$, e $B_i = Y_i - (a + b.X_i)$, logo tem-se:

$$Y_i - (a + b.X_i) - (Y_{i+1} - (a + b.X_{i+1})) \leq \theta_i \quad (\text{Eq. 51})$$

e

$$-(Y_i - (a + b.X_i)) + Y_{i+1} - (a + b.X_{i+1}) \leq \theta_i \quad (\text{Eq. 52})$$

Estas inequações podem ainda ser simplificadas, perdendo-se o parâmetro a :

$$Y_i - a - b.X_i - Y_{i+1} + a + b.X_{i+1} \leq \theta_i \quad (\text{Eq. 53})$$

\Leftrightarrow

$$Y_i - Y_{i+1} - b(X_i - X_{i+1}) \leq \theta_i \quad (\text{Eq. 54})$$

e

$$-Y_i + a + b.X_i + Y_{i+1} - a - b.X_{i+1} \leq \theta_i \quad (\text{Eq. 55})$$

\Leftrightarrow

$$-Y_i + Y_{i+1} + b(X_i - X_{i+1}) \leq \theta_i \quad (\text{Eq. 56})$$

Trabalhando um pouco mais, tem-se:

$$b(X_{i+1} - X_i) - \theta_i \leq Y_i - Y_{i+1} \quad (\text{Eq. 57})$$

e

$$b(X_i - X_{i+1}) - \theta_i \leq Y_{i+1} - Y_i \quad (\text{Eq. 58})$$

Observou-se ainda a restrição $\theta_i \geq 0$, que também tem de ser trabalhada de modo a cumprir o postulado pela função *linprog*, e como tal passa a ser:

$$-\theta_i \leq 0 \quad (\text{Eq. 59})$$

Finalmente, outra das restrições a ter em conta é $B_{non-Stemming} \geq B_{ideal}$, o que se traduz numa inequação:

$$-Y_i + a + b.X_i \leq -B_{ideal} , \text{ para } i=1 \text{ até } n \quad (\text{Eq. 60})$$

\Leftrightarrow

$$a + b.X_i \leq B_{ideal} + Y_i \quad (\text{Eq. 61})$$

Assumindo estes pressupostos, é possível formar um sistema de inequações para $i=1$ até aos n pontos com que o algoritmo está a operar tanto no perfil crítico como no furo:

$$\left\{ \begin{array}{l} b(X_{i+1} - X_i) - \theta_i \leq Y_i - Y_{i+1} \\ b(X_i - X_{i+1}) - \theta_i \leq Y_{i+1} - Y_i \\ -\theta_i \leq 0 \\ (...) \\ b(X_n - X_{n-1}) - \theta_{n-1} \leq Y_{n-1} - Y_n \\ b(X_{n-1} - X_n) - \theta_{n-1} \leq Y_n - Y_{n-1} \\ -\theta_{n-1} \leq 0 \\ a + b.X_i \leq B_{ideal} + Y_i \\ a + b.X_{i+1} \leq B_{ideal} + Y_{i+1} \\ (...) \\ a + b.X_n \leq B_{ideal} + Y_n \end{array} \right. \quad (\text{Eq. 62})$$

Este sistema é convertido num sistema matricial³ composto por:

- Matriz A , composta pelos coeficientes das variáveis do sistema, e tem dimensão $m \times n$, sendo m , o número total de inequações que restringem o problema, e n o número de variáveis associadas. A cada ponto discretizado do furo estão sempre associadas 4 restrições (3 relativas ao parâmetro $\theta_i + 1$ relativa ao B_{ideal});
- Vetor x , formado pelas variáveis de decisão a , b , θ_i (...) e θ_{n-1} , de dimensão $n \times 1$
- Vetor b , constituído pelos termos independentes, de dimensão $m \times 1$

$$A = \begin{bmatrix} 0 & X_{i+1} - X_i & -1 & \cdots & 0 \\ 0 & X_i - X_{i+1} & -1 & \cdots & 0 \\ 0 & 0 & -1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & X_n - X_{n-1} & 0 & \cdots & -1 \\ 0 & X_{n-1} - X_n & 0 & \cdots & -1 \\ 0 & 0 & 0 & \cdots & -1 \\ 1 & X_i & 0 & \cdots & 0 \\ 1 & X_{i+1} & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & X_n & 0 & \cdots & 0 \end{bmatrix} \quad (\text{Eq. 63})$$

$$x = \begin{bmatrix} a \\ b \\ \theta_i \\ \vdots \\ \theta_{n-1} \end{bmatrix} \quad (\text{Eq. 64})$$

$$b = \begin{bmatrix} Y_i - Y_{i+1} \\ Y_{i+1} - Y_i \\ 0 \\ \vdots \\ Y_{n-1} - Y_n \\ Y_n - Y_{n-1} \\ 0 \\ B_{ideal} + Y_i \\ B_{ideal} + Y_{i+1} \\ \vdots \\ B_{ideal} + Y_n \end{bmatrix} \quad (\text{Eq. 65})$$

³ Ver subcapítulo 3.2.2.2.

No que diz respeito à programação em MATLAB para obtenção destas entidades, é necessário que o algoritmo leia o número de pontos que constituem o perfil crítico (já sem os pontos do tampão), e que faça a construção de A e b a partir desse número de pontos, usando para o efeito matrizes nulas com uma dada dimensão.

Tomando l como o número de pontos que constituem o vetor do perfil crítico (número de linhas desse vetor), a matriz A terá dimensão $m \times n$, sendo $m = (l-1) \times 4 + 1$ e $n = (l-1)+2$, e o vetor b terá dimensão $m \times 1$.

```

330 % algoritmo de otimização
331
332 % formulação da matriz A e vetor B
333
334 [m n] = size(dataSolver);
335
336 m=m-1;
337
338
339 A=zeros(m*4+1,m+2);
340 B=zeros(m*4+1,1);
341 index=1;
342
343 for i=1:m
344     A(index,2)= dataSolver(i+1,1)-dataSolver(i,1);
345     A(index+1,2)= dataSolver(i,1)-dataSolver(i+1,1);
346     A(index,i+2)= -1;
347     A(index+1,i+2)= -1;
348
349     B(index,1)=dataSolver(i+1,2)-dataSolver(i,2);
350     B(index+1,1)=dataSolver(i,2)-dataSolver(i+1,2);
351
352     index=index+2;
353 end
354
355 for i=1:m
356     A(index,2+i)= -1;
357
358     B(index,1)=0;
359
360     index=index+1;
361 end
362
363 for i=1:m+1
364     A(index,1)= 1;
365     A(index,2)= dataSolver(i,1);
366
367     B(index,1)=-maxBurden+dataSolver(i,2);
368
369     index=index+1;
370 end

```

Figura 50 – Excerto de código relativo à formulação da matriz A e vetor b

4.3.6.3. Formulação da função objetivo

Tendo como dado adquirido anteriormente, que a função objetivo deste problema é dada por *Função objetivo* = $\min\{\sum \theta_i + \sum B_i\}$, então:

$$\text{Função objetivo} = \min\{\sum \theta_i + \sum B_i\} \quad (\text{Eq. 66})$$

$$\Leftrightarrow$$

$$\text{Função objetivo} = \min\{(\theta_i + \theta_{i+1} + \dots + \theta_{n-1}) + (B_i + B_{i+1} + \dots + B_{n-1})\} \quad (\text{Eq. 67})$$

$$\Leftrightarrow$$

$$\begin{aligned} \text{Função objetivo} = \min\{ & (\theta_i + \theta_{i+1} + \dots + \theta_{n-1}) + ((Y_i - a - b * X_i) + (Y_{i+1} - a - b * X_{i+1}) + \dots \\ & + (Y_{n-1} - a - b * X_{n-1})) \} \end{aligned} \quad (\text{Eq. 68})$$

A função objetivo é representada por um vetor f , cujo *input* deverá ser feito à semelhança da matriz A e do vetor b , isto é, a função é obtida com recurso a um vetor-linha nulo, com comprimento $m + 2$ (sendo m , o número de inequações de restrição, visto no ponto anterior). Os elementos que compõem o vetor f são definidos à custa do vetor x , e como tal terão correspondência com as variáveis de decisão do problema. O primeiro elemento de f corresponde ao valor simétrico do número de variáveis de decisão a do problema. O segundo elemento corresponde ao somatório de X_i que multiplica pela variável de decisão b . Os restantes elementos de $i=3$ até m correspondem assumem valor 1, pois cada um dos elementos multiplica pela respetiva variável de decisão θ_i .

```

385      % definir soma de Xi, para multiplicação por B
386
387      B=B';
388
389      somaX=0;
390
391      for i=1:m+1
392          somaX=somaX+dataSolver(i,1);
393      end
394
395      %função objetivo
396
397      f=zeros(1,m+2);
398
399      f(1,1)=-(m+1);
400      f(1,2)=-somaX;
401
402
403      for i=1:m
404          f(1,i+2)=1;
405      end

```

Figura 51 – Formulação do vetor f , relativo à função objetivo

4.3.6.4. Resultado final da otimização

A solução final para a minimização da função objetivo é obtida como uma variável de saída nomeada *solution*, que corresponde ao vetor x definido anteriormente, cujo comprimento será n . Através do valor dos elementos x_1 e x_2 do vetor (respetivamente as variáveis de decisão a e b), é possível obter a equação da reta que representa o furo otimizado.

```

407 % resolução do problema com linprog
408
409 [solution,fval,exitflag,output] = linprog(f,A,B);
410 clc;
411
412 % matriz de resultados
413 result=[result;[hole , criticalBurden(1,1)-solution(1)+solution(2)*criticalBurden(1,1) , ...
414             atan(solution(2))/pi*180]];
415
416 hold on
417 line(criticalBurden(:,1),criticalBurden(:,1)*solution(2)+solution(1), 'Color', 'red')
418
419 end
420
421 result

```

Figura 52 – Excerto de algoritmo relativo à programação linear com função *linprog* e variáveis de saída

O *output* final do algoritmo de otimização é uma matriz denominada *result*, de dimensão $n \times 3$, em que n é o número de furos dados como *input* inicial ao algoritmo. Na primeira coluna, estão os índices relativos aos furos (de $i=1$ até n), na segunda coluna estão os dados relativos ao afastamento na crista do furo, e na terceira coluna estão colocados os valores de inclinação do furo, que o próprio algoritmo calcula através do parâmetro b da equação da reta:

$$\text{Inclinação (}^\circ\text{)} = \frac{\tan^{-1}(b)}{\pi} * 180 \quad (\text{Eq. 69})$$

result =		
1.0000	4.2046	22.6068
2.0000	4.9875	25.8625
3.0000	4.1524	18.1920
4.0000	3.5935	11.1584
5.0000	3.4220	3.5928
6.0000	3.9571	7.6958
7.0000	4.6319	17.2877
8.0000	3.7718	15.6593
9.0000	3.0898	3.4601
10.0000	2.6871	0.1507
11.0000	3.8673	9.9404

Figura 53 – Exemplo de *output* da matriz *result* para 11 furos

O ficheiro faz ainda a representação gráfica de todos os furos e dos respetivos perfis críticos associados. Assim, para o exemplo de *output* dado na Figura 53, são obtidas onze figuras que dizem respeito ao perfil crítico em três dimensões, e outras onze figuras que representam o perfil crítico e a reta do furo em duas dimensões. Abaixo, encontram-se dois desses exemplos:

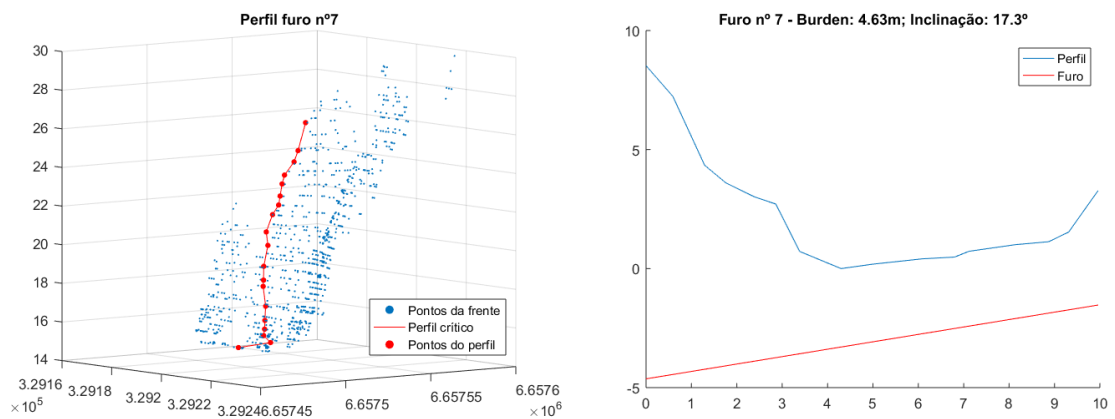


Figura 54 - Perfil crítico nº7 (representação 3D, à esquerda) e perfil e furo nº7 (representação 2D, à direita)

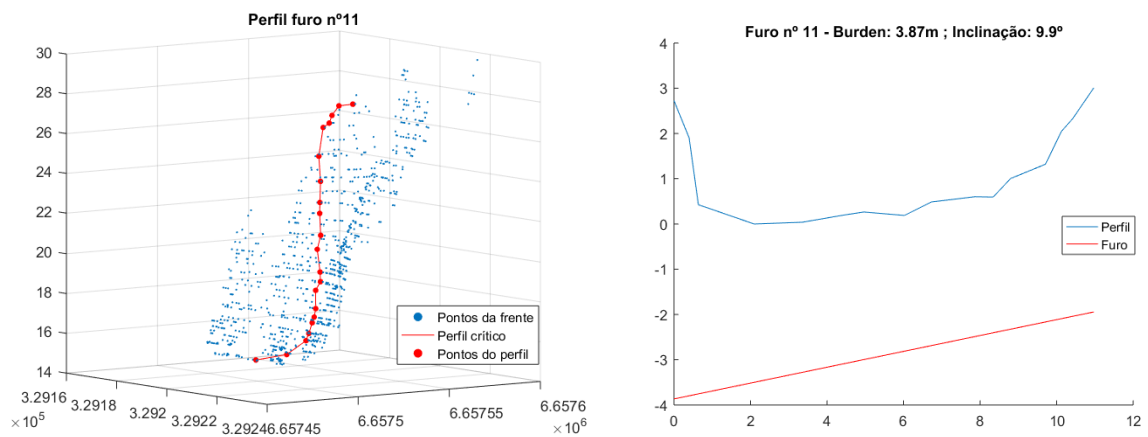


Figura 55 – Perfil crítico nº11 (representação 3D, à esquerda) e perfil e furo nº11 (representação 2D, à direita)

No anexo B encontram-se dados mais detalhados de utilização do algoritmo relativamente a dois exemplos de frente livre com o dimensionamento da primeira fila de furos.

4.4. MODELAÇÃO E RESOLUÇÃO DO PROBLEMA UTILIZANDO PROGRAMAÇÃO LINEAR NO EXCEL

Uma vez obtidos os dados, é necessário formular o problema, tendo em atenção os pressupostos observados no capítulo 3 do presente trabalho. O objetivo é minimizar, de um modo uniforme, a distância dos pontos do furo aos pontos da frente livre sem romper as restrições de afastamento mínimo.

4.4.1. CONSTRUÇÃO DO MODELO DO FURO EM EXCEL

4.4.1.1. Preparação prévia de amostra de pontos

Para obter o perfil crítico no Excel é necessário importar os dados que deverão já estar previamente tratados pelo algoritmo MATLAB visto anteriormente.

O *Solver* tem algumas limitações, e surgem alguns problemas quando é necessário trabalhar com um número significativo de pontos, podendo não devolver uma solução ótima, ou até mesmo não funcionar de todo, uma vez que esta ferramenta só consegue resolver problemas limitados a 200 células de variável e 100 restrições.

Como tal, é necessário considerar uma amostra de pontos mais reduzida, e optou-se por eliminar os pontos com índice par, reduzindo a amostra para apenas 21 pontos. Ao correr o *solver*, persistiram alguns problemas de otimização, e como tal, eliminaram-se 6 pontos de modo intercalado, de modo a obter uma amostra de 15 pontos que seja representativa, e que seja também mais leve, de maneira a que o *Solver* seja capaz de devolver uma solução ótima adequada.

i	X	Y	i	X	Y	i	X	Y
1	0	0	1	0	0	1	0	0
2	0,574	1,59151	2	0,837	2,22696	2	0,837	2,22696
3	0,837	2,22696	3	1,417	3,02324	3	1,417	3,02324
4	1,141	2,82375	4	2,18	3,92183	4	3,581	5,51244
5	1,417	3,02324	5	3,581	5,51244	5	6,913	6,02836
6	2,014	3,71656	6	6,913	6,02836	6	10,383	7,77622
7	2,18	3,92183	7	9,475	6,89538	7	11,414	8,04799
8	3,365	4,60936	8	10,383	7,77622	8	12,174	10,0827
9	3,581	5,51244	9	11,414	8,04799	9	13,154	10,9357
10	5,538	5,48515	10	11,794	8,67197	10	14,536	12,5333
11	6,913	6,02836	11	12,174	10,0827	11	15,386	14,2315
12	8,219	6,04747	12	13,154	10,9357	12	16,683	15,5571
13	9,475	6,89538	13	13,419	11,1829	13	17,143	16,0246
14	10,146	7,27963	14	14,536	12,5333	14	17,973	16,5196
15	10,383	7,77622	15	15,386	14,2315	15	19,213	16,8037
16	10,981	7,9525	16	15,774	14,4572	16	20,094	17,4305
17	11,414	8,04799	17	16,683	15,5571			
18	11,645	8,46444	18	17,143	16,0246			
19	11,794	8,67197	19	17,609	16,3446			
20	11,913	10,0513	20	17,973	16,5196			
21	12,174	10,0827	21	19,213	16,8037			
22	12,798	10,8811	22	20,094	17,4305			
23	13,154	10,9357						
24	13,17	11,0761						
25	13,419	11,1829						
26	14,161	12,1043						
27	14,536	12,5333						
28	15,034	13,3108						
29	15,386	14,2315						
30	15,728	14,3446						
31	15,774	14,4572						
32	16,201	15,0154						
33	16,683	15,5571						
34	16,907	15,8827						
35	17,143	16,0246						
36	17,271	16,1635						
37	17,609	16,3446						
38	17,774	16,5004						
39	17,973	16,5196						
40	18,282	16,6718						
41	19,213	16,8037						
42	19,564	17,0972						
43	20,094	17,4305						
44	20,459	21,7497						

Figura 56 – Exemplo de redução da amostra de pontos no Microsoft Excel

i	X	Y	Y Furo	Bi	Delta Burden
1	0	0	0	0	-2,226962057
2	0,837	2,226962	0	2,226962	-0,796275674
3	1,417	3,023238	0	3,023238	-2,489206725
4	3,581	5,512444	0	5,512444	-0,515912669
5	6,913	6,028357	0	6,028357	-1,747861103
6	10,383	7,776218	0	7,776218	-0,271773339
7	11,414	8,047992	0	8,047992	-2,034735265
8	12,174	10,08273	0	10,08273	-0,852935439
9	13,154	10,93566	0	10,93566	-1,597624256
10	14,536	12,53329	0	12,53329	-1,698204763
11	15,386	14,23149	0	14,23149	-1,325655638
12	16,683	15,55715	0	15,55715	-0,467443931
13	17,143	16,02459	0	16,02459	-0,494993393
14	17,973	16,51958	0	16,51958	-0,284081324
15	19,213	16,80367	0	16,80367	-0,626874748
16	20,094	17,43054	0	17,43054	

Figura 57 – Tabela-padrão para inserção de dados

4.4.1.2. Modelo do furo e afastamento crítico

Tendo uma amostra representativa de pontos correspondentes ao perfil da frente livre, é possível prosseguir com a construção do modelo. O próximo passo é programar uma coluna que corresponderá ao modelo matemático do furo $y=ax+b$, sendo a variável x coincidente com os valores de x da frente, e ainda duas células correspondentes aos parâmetros a e b que, como já foi mencionado anteriormente, serão variáveis de decisão deste modelo.

Constroem-se ainda duas colunas, referentes aos parâmetros B_i e ΔB_i , lembrando que B_i representa o afastamento de cada ponto do furo em relação à frente livre, ao longo de todo o furo, naturalmente obtido fazendo a diferença entre as ordenadas de cada ponto da frente e do furo:

$$B_i = y_{frente} - y_{furo} \quad (\text{Eq. 70})$$

E que ΔB_i é um parâmetro relativo à diferença de valor entre sucessivos afastamentos B , entre os pontos discretizados do furo e os pontos da frente livre:

$$\Delta B_1 = B_1 - B_2 \quad (\text{Eq. 71})$$

$$\Delta B_2 = B_2 - B_3 \quad (\text{Eq. 72})$$

(...)

$$\Delta B_{n-1} = B_{n-1} - B_n \quad (\text{Eq. 73})$$

Mais uma vez, dado que a zona do tampão não tem carga explosiva, optou-se por não considerar os pontos correspondentes ao tampão, e proceder à otimização a partir dos pontos discretizados imediatamente a seguir. Na folha de cálculo, pontos não-considerados estão identificados como *stemming* a cor vermelha.

4.4.2. VARIÁVEIS DE DECISÃO, FUNÇÃO OBJETIVO E RESTRIÇÕES

As variáveis de decisão são as mesmas consideradas no algoritmo do MATLAB: a , b e θ_i , sendo θ_i :

$$|\Delta B_i| \leq \theta_i \rightarrow -\theta_i \leq \Delta B_i \leq \theta_i \quad (\text{Eq. 74})$$

A função objetivo é novamente a minimização da soma dos valores-limite, $\sum \theta_i$, com a soma dos valores do afastamento, $\sum B_i$, ou seja:

$$\text{Função objetivo} = \min\{\sum \theta_i + \sum B_i\} \quad (\text{Eq. 75})$$

À semelhança do que sucede no algoritmo de otimização em MATLAB, foram consideradas as seguintes restrições, com a adição de uma quarta restrição:

$B_{non-Stemming} \geq B_{ideal}$, restringe todos os pontos no intervalo $y =]final\ do\ tampão,\ pé\ do\ furo]$, de modo a que o seu afastamento em relação à face livre seja igual ou superior ao afastamento crítico definido;

- $-\theta_i \leq \Delta B_i \leq \theta_i$, que se desdobra em duas restrições:
 - $\Delta B_i \leq \theta_i$, restringe todas as diferenças sucessivas de afastamento a serem iguais ou inferiores aos valores-limite admitidos pelo *Solver*;
 - $\Delta B_i \geq -\theta_i$, restringe todas as diferenças sucessivas de afastamento a serem iguais ou superiores ao simétrico dos valores-limite admitidos pelo *Solver*;
- $\theta_i \geq 0$, restringe todos os valores θ_i a valores não-negativos
- $b \geq 0$, restringe o parâmetro b (declive da reta que modela o furo) a valores não-negativos, de modo a que a reta possa estar representada paralelamente ao perfil da frente. Valores negativos fariam com que a reta fosse representada espelhada em relação ao semi-eixo positivo xx .

4.4.3. SOLUÇÃO ÓTIMA APRESENTADA PELO ALGORITMO

Tendo a folha de cálculo do Excel toda preparada, basta abrir o *Solver* e atribuir a cada parâmetro (função objetivo, variáveis de decisão e restrições), as células respetivas da folha de cálculo. Para a resolução do problema, é necessário desseleccionar a caixa «Tornar não negativas variáveis não constringidas», de modo a que a variável de decisão a possa tomar valores negativos (embora todas as outras não possam), e seleccionar como método de resolução o algoritmo LP Simplex, como é visível na figura Figura 58.

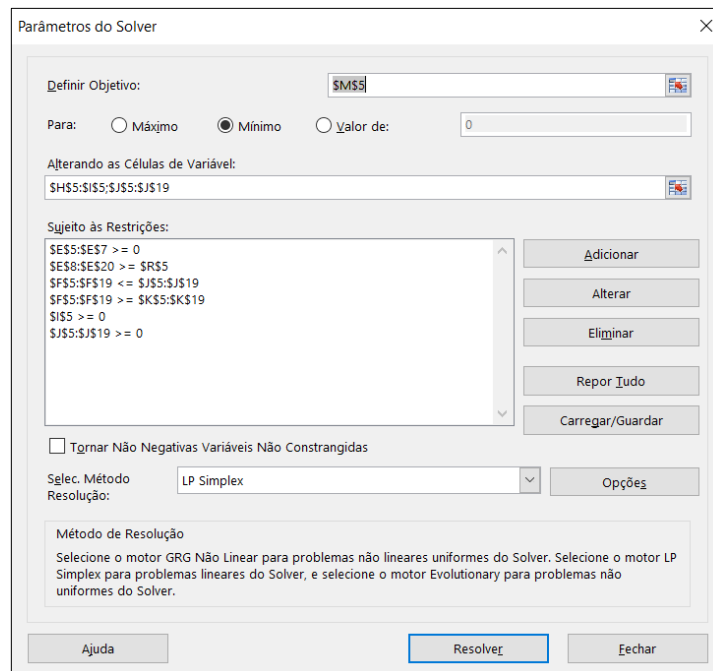


Figura 58 – Inserção de parâmetros no *Solver*

Após clicar em «Resolver», o *Solver* calcula a solução ótima, e as células referentes às variáveis de decisão mudam instantaneamente de valor, assumindo os valores ótimos encontrados pelo algoritmo. Os resultados obtidos para o exemplo da frente livre com a nuvem de pontos da figura (numerar figura) foram os seguintes:

						Variáveis de Decisão				
i	X	Y	Y Furo	Bi	Delta Burden		b	a	θ_i	$-\theta_i$
1	0	0	-3,07362974	3,07363	-1,63139547	stemming	-3,073629737	0,711549	1,631395	-1,63139547
2	0,837	2,226962	-2,47806315	4,705025	-0,383577203	stemming			0,383577	-0,383577203
3	1,417	3,023238	-2,06536468	5,088602	-0,949414496	stemming			0,949414	-0,949414496
4	3,581	5,512444	-0,52557245	6,038017	1,854968897				1,854969	-1,854968897
5	6,913	6,028357	1,845309117	4,183048	0,721214237				0,721214	-0,721214237
6	10,383	7,776218	4,314384457	3,461834	0,461833772		Função Objetivo		0,461834	-0,461833772
7	11,414	8,047992	5,047991568	3	-1,493957957		$\sum \theta_i$		1,493958	-1,493957957
8	12,174	10,08273	5,588768876	4,493958	-0,155617332		93,78841056		0,155617	-0,155617332
9	13,154	10,93566	6,286086984	4,649575	-0,614263415				0,614263	-0,614263415
10	14,536	12,53329	7,269447825	5,263839	-1,093388037				1,093388	-1,093388037
11	15,386	14,23149	7,874264551	6,357227	-0,40277647				0,402776	-0,40277647
12	16,683	15,55715	8,79714372	6,760003	-0,14013135				0,140131	-0,14013135
13	17,143	16,02459	9,124456301	6,900135	0,095592351				0,095592	-0,095592351
14	17,973	16,51958	9,715042045	6,804542	0,598239546				0,59824	-0,598239546
15	19,213	16,80367	10,59736292	6,206303	0				2,78E-17	0
16	20,094	17,43054	11,22423766	6,206303						

Figura 59 – Coordenadas da frente livre, definição da reta do furo, parâmetros de afastamento, função objetivo minimizada e variáveis de decisão otimizadas pelo Solver

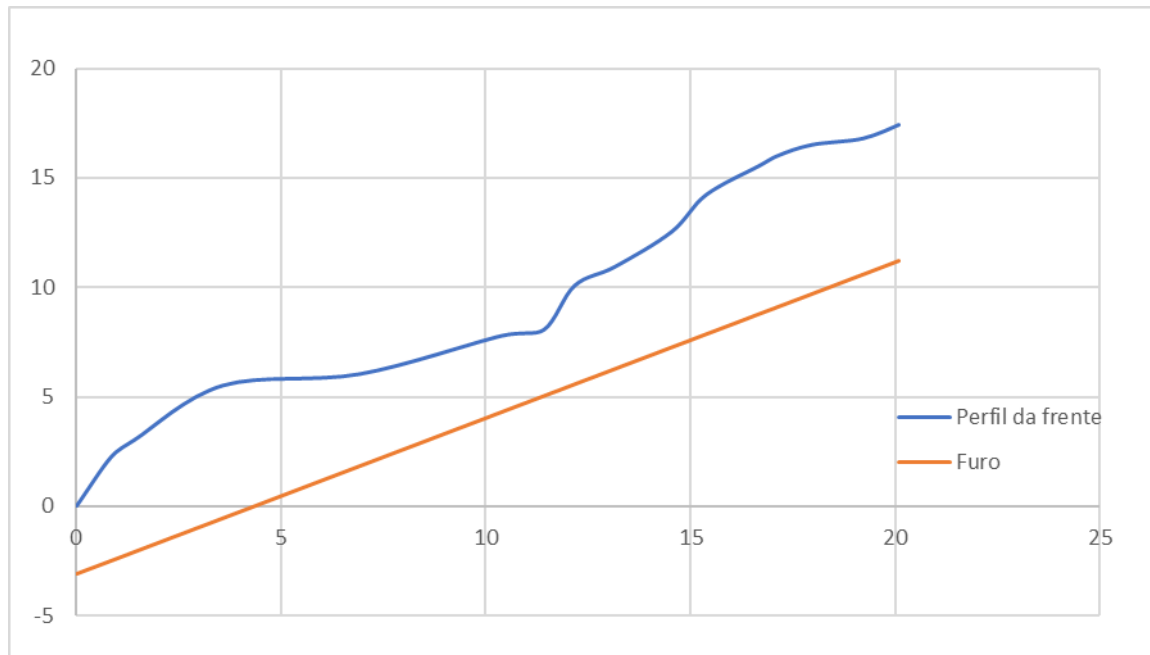


Figura 60 – Representação gráfica em Excel, com o furo otimizado

A partir desta solução, é possível calcular a inclinação do furo em graus, e também é possível calcular outros parâmetros de fogo, como o afastamento B da boca do furo à crista da frente livre, a altura H da frente livre ou o comprimento L do furo, tendo em atenção que a subfuração J deverá assumir um valor de 1m para este exemplo:

- $Inclinação (^{\circ}) = \frac{\tan^{-1}(a)}{\pi} * 180$, sendo b , o declive da reta representativa do furo;
- $B = B_1$, sendo B_1 , o afastamento do ponto $i=1$ do furo em relação ao ponto $i=1$ da frente;
- $H = \max(x) - \min(x)$, em que x diz respeito aos valores das abcissas tanto da frente como do furo;
- $L = \tan(Inclinação) * H$, devendo o valor da inclinação ser convertido em radianos.

Os resultados obtidos para esta solução são os indicados abaixo:

Subdrilling (m)	1
Hole length (m)	15.30
Bench height (m)	20.09
Burden (m)	3.07
Inclination ($^{\circ}$)	35.4

Figura 61 – Quadro de parâmetros

Verifica-se que para esta frente livre, a inclinação ótima deverá ser de cerca de 35.4° , com um comprimento de furo de cerca de 15.30 m, subfuração 1 m e o afastamento da boca do furo à crista da bancada cerca de 3.07 m.

Nenhum dos valores encontrados pelo algoritmo é inferior ao afastamento ideal definido como 3 m, como era pretendido.

5

SUGESTÕES PARA TRABALHOS FUTUROS COM ELABORAÇÃO DE CASO DE ESTUDO

5.1. NOTA PRÉVIA

No capítulo anterior, em simultâneo com a formulação do modelo, foi possível observar alguns dos resultados obtidos pelo algoritmo utilizando dados de levantamentos topográficos já existentes, com a indicação dos valores de afastamento e da inclinação que deveriam ser utilizados naqueles casos. De modo a fazer a validação deste modelo, é necessário testá-lo em ambiente de campo, fazendo o levantamento dos dados de topografia da frente livre, e executando um desmonte real com os parâmetros ajustados pelo algoritmo. Uma vez que, por diversos motivos, não foi possível realizar essa experimentação durante o decurso deste trabalho, neste capítulo tecem-se algumas considerações sobre o procedimento experimental que deverá ser efetuado para estudar o comportamento do modelo em campo.

5.2. DESCRIÇÃO DO PROCEDIMENTO EXPERIMENTAL A DESENVOLVER

5.2.1. SELEÇÃO DA BANCADA A DESMONTAR

Uma vez definidos todos os parâmetros finais da pega de fogo, poderá ser realizado o desmonte da bancada em estudo. Como nota prévia, e para que os resultados sejam analisados da melhor forma, o ideal será desmontar uma bancada o mais homogênea possível em termos de irregularidades geométricas da frente, crista e pé, e da fragmentação do maciço, de modo a que não haja demasiada heterogeneidade entre zonas da bancada, que levem a que o dimensionamento da pega de fogo seja afetado em função disso.

A bancada deverá ser ainda comprida o suficiente para dividir a pega de fogo em duas pegas, uma metade com os furos otimizados um a um, de acordo com o algoritmo, e outra metade sem otimizar os furos, utilizando uma inclinação padronizada para os furos, como seria feito sem recorrer ao algoritmo. Poderia ainda ser considerada a comparação de um desmonte total com a primeira linha de furos otimizada, com outros desmontes em que o algoritmo não seja utilizado. No entanto, poderia haver alta variabilidade entre as bancadas desmontadas a nível da sua dimensão, geometria e fragmentação, e isso faria com que não fosse possível a comparação entre desmontes.

5.2.2. LEVANTAMENTO TOPOGRÁFICO DA FRENTE LIVRE

Uma vez selecionada a bancada a desmontar, é necessário adquirir os dados relativos à frente livre. Este processo pode ser feito com recurso a dois tipos de equipamento: perfilómetro laser ou drone equipado com câmara de alta definição. Em ambas as situações, é de extrema importância que a frente livre esteja completamente limpa de fragmentos de desmontes anteriores, de modo a que os equipamentos consigam recolher nuvens de pontos que permitam gerar um modelo da frente livre o mais aproximado da realidade.

5.2.2.1. Levantamento topográfico com recurso a perfilómetro laser

A utilização de um perfilómetro laser permite obter um modelo detalhado de uma dada região da bancada: frente livre, furos, crista e pé. O equipamento consegue fazer o levantamento de todos estes dados, em leituras separadas, associando-lhes um código numérico de dois dígitos (à escolha do operador).

Seguindo o manual de instruções fornecido com o equipamento (que é variável entre os diversos modelos no mercado), deverá ser necessário inserir alguns parâmetros no perfilómetro para que este possa efetuar a leitura correta da frente, dando como exemplo:

- Altura a que o equipamento dista do solo;
- Altura do alvo – para recolha de pontos isolados no topo da bancada, como é o caso dos furos;
- Distância mínima e máxima de medição – deverão ser valores coerentes com a distância a que o equipamento dista da frente livre, de modo a não captar pontos de terreno que não façam parte da frente livre;
- Intervalo de medição – para obter uma frente livre o mais detalhada possível, deverá ser inserido um intervalo pequeno (< 0.35 m) e deverá ser feito o scan horizontal e vertical da frente.



Figura 62 – Levantamento topográfico com perfilómetro laser

Os lasers que fazem *scan* automático podem ainda requerer a captura manual de dois pontos extremos da frente livre (ponto inferior esquerdo e ponto superior direito) e, selecionando a forma aproximada da frente (poligonal ou retangular), o equipamento fará a leitura automática de todos os pontos contidos nos intervalos de medição e de distância definidos previamente. É aconselhável visar os pontos relativos à crista e ao pé da bancada, no entanto é possível desenhar linhas relativas a estes parâmetros na plataforma O-Pitblast®.

Para a execução do algoritmo não é obrigatório o levantamento da posição dos furos, uma vez que as posições podem ser inseridas manualmente pelo operador. Contudo, caso se pretenda tirar a posição da crista dos furos com recurso ao laser, basta inserir a altura do alvo e, inserindo um novo código numérico relativo a esta leitura, fazer a picagem manual dos pontos relativos aos furos.

5.2.2.2. Obtenção de dados para execução do algoritmo

De modo a obter os dados e fazer o modelo da bancada, é necessário conectar o laser a um computador através de um cabo USB, e fazer a descarga dos dados utilizando um *software* adequado à transmissão de dados do equipamento para o computador (por exemplo, o *software* PuTTY). Inserindo o código que é obtido pelo PuTTY num bloco de notas do Windows, e mudando a extensão do ficheiro para CDU, é possível obter um modelo da bancada através da plataforma O-Pitblast®. Para tal, bastará aceder ao menu *Free-Face*, selecionar a opção *Import Free-Face* e selecionar o ficheiro CDU correspondente, tomando cada coluna uma das coordenadas: x , y e z .

Finda esta etapa de importação do modelo da frente, o operador poderá verificar se o modelo é uma representação fidedigna da realidade, e caso seja um modelo adequado, pode proceder à exportação das coordenadas x , y e z dos pontos da frente livre para uma folha de cálculo que permita a leitura do MATLAB, conforme foi visto no capítulo 4. Para fazer a exportação é necessário aceder ao menu *Topography* e selecionar a opção *Export → Terrain*. Assim, será obtido um ficheiro CSV que permitirá fazer o *input* das coordenadas da frente no MATLAB.

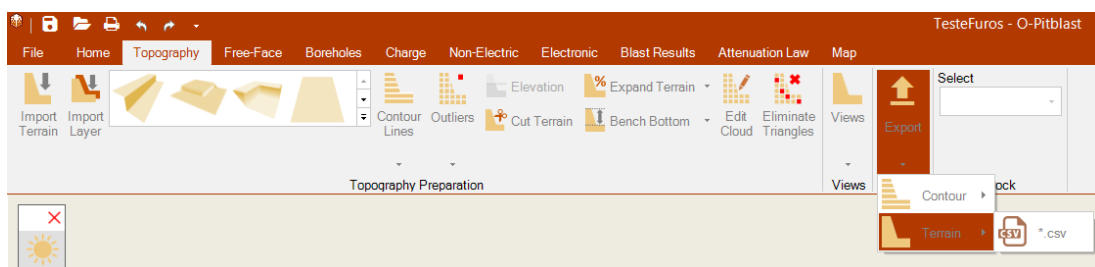


Figura 63 – Exportação do modelo para ficheiro CSV

Caso tenham sido obtidos pontos relativos aos furos, é possível importá-los de modo idêntico à importação da frente livre, bastando para o efeito associar o código utilizado durante o levantamento dos pontos. Caso não tenham sido obtidos em campo, é possível adicionar os furos desejados no menu *Boreholes*. É ainda possível exportar um ficheiro CSV relativo aos furos (*Boreholes → Export → .csv*), lembrando que no caso da execução deste algoritmo em MATLAB, basta apenas exportar dados relativos à primeira linha de furos.

5.2.2.3. Levantamento topográfico com recurso a drone equipamento com câmara de alta resolução

A utilização de drones na indústria mineira tem vindo a ganhar bastante espaço nos últimos anos, uma vez que são equipamentos extremamente versáteis, que permitem não só fazer a modelação de uma bancada, ou de uma exploração inteira, como também permitem fazer, por exemplo, a medição de gases de explosão e deteção de NOx, ou análise granulométrica com recurso a fotogrametria.



Figura 64 – Levantamento topográfico com drone

No caso do levantamento topográfico, basta levantar o drone e percorrer toda a frente livre, capturando um número adequado de imagens para posterior modelação da bancada. O modelo é obtido através da sobreposição das várias fotografias georreferenciadas, permitindo construir um modelo 3D com recurso a um *software* como o *Agisoft Photo Scan*. Após obter o modelo em Agisoft, é possível importar a bancada para a plataforma O-Pitblast® e obter os dados da frente livre para execução do algoritmo de maneira idêntica ao que foi descrito com recurso ao perfilómetro laser. No entanto, é necessário ter em conta que, uma vez que é obtido um modelo de uma bancada inteira, este deverá ser ajustado de modo a que seja obtida uma nuvem de pontos que somente diga respeito à frente livre, e como tal é necessário fazer uso da ferramenta de corte do O-Pitblast® para exportar apenas a parte do terreno que corresponde à frente livre.

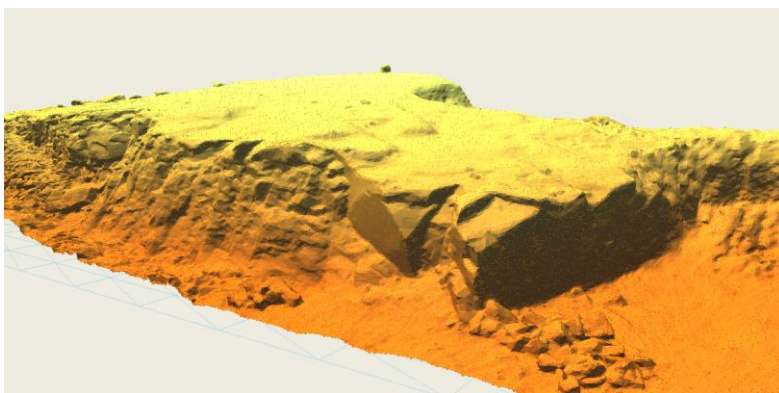


Figura 65 - Modelo de bancada gerado com recurso a drone

Durante o levantamento com recurso ao drone é necessário ter alguns cuidados, nomeadamente:

- Ter em consideração a direção e velocidade dos ventos, que poderão desestabilizar o drone ou até impedir o seu voo;
- Ter cuidado com a presença de aves, nomeadamente gaivotas e aves de rapina, que são extremamente territoriais, tendo já havido inúmeras ocorrências de ataques de aves a drones, provocando alguns danos ou até mesmo a sua queda;
- Ter em atenção a presença de dumpers e outros equipamentos de trabalho que possam intercalar a trajetória do drone ou provocar fluxos de ar repentinos que originem a sua queda;
- Ter em consideração a luz solar, pois é fundamental ter boas condições de iluminação para poder obter fotos nítidas do terreno, caso contrário o modelo poderá ficar com falhas devido à presença de sombras na bancada.

5.2.3. EXECUÇÃO DO ALGORITMO

Uma vez obtidos os dados da frente livre e dos furos numa folha de cálculo (CSV), prossegue-se então para a execução do algoritmo, tal e qual como foi visto no capítulo 4 desta Dissertação. Após a sua execução, são obtidos os valores otimizados relativos à inclinação e afastamento na boca dos furos, que deverão ser posteriormente fornecidos ao perfurador. Para melhor visualização dos resultados, aconselha-se a fazer uma segunda importação do modelo da frente livre e dos dados dos furos pós-otimização para a plataforma O-Pitblast®, que poderá auxiliar na discussão crítica dos resultados obtidos, e ajudar a visualizar se o comportamento dos furos relativamente à frente livre é o esperado.

Face aos resultados obtidos, poderá ser necessário fazer alguns ajustes no dimensionamento da pega de fogo planeada anteriormente, nomeadamente ao nível dos afastamentos entre filas, espaçamento entre furos, comprimento dos furos e subfuração, ou até mesmo na quantidade de explosivo a utilizar e nos tempos de iniciação da pega de fogo.

Admite-se ainda poder ser necessário proceder a alguns ajustes no próprio algoritmo, caso os resultados não coincidam com o que é esperado e previsto.

5.2.4. PERFURAÇÃO E VERIFICAÇÃO COM INSTRUMENTO MEDIDOR DE DESVIO DE FUROS

Fornecendo os dados ao perfurista, este deverá ser capaz de efetuar a perfuração de maneira mais precisa possível, para evitar os desvios dos furos. Por vezes essa tarefa torna-se bastante difícil devido a falhas no equipamento de perfuração, sujeito a bastante desgaste, ou devido a fatores inerentes ao maciço, como a fraturação, que pode influenciar bastante o desvio dos furos em relação ao planeado.

Para fazer a verificação da inclinação dos furos, torna-se bastante útil o recurso a um instrumento medidor de desvio de furos, como é o caso do Boretrak. Estes equipamentos estão disponíveis em dois formatos:

- Boretrak de barras: mais preciso, permite a medição em ambientes subterrâneos, áreas de material ferroso, atingindo grandes profundidades. Não é tão prático como a versão de cabo, mas permite aplicações muito mais variadas.
- Boretrak de cabo: mais ágil, compacto, leve e resistente, permite fazer medições em cavidades com até 65m de profundidade, dependendo de bússola para efetuar a medição do ângulo, não é adequado para ambientes ferrosos.

A vantagem do controlo do desvio dos furos reflete-se bastante nos resultados do desmonte, principalmente no que diz respeito à fragmentação e à projecção de blocos. É sabido que na maioria das vezes os desmontes diferem do planeado, nomeadamente no que concerne a inclinação dos furos, e a utilização deste tipo de equipamento permite tomar medidas para evitar que o desmonte seja descontrolado e provoque efeitos indesejáveis.



Figura 66 - Boretrak® de barras (<https://optron.com/carlson/products/rodde-boretrak/>)

Após a utilização do Boretrak® no terreno, é possível importar o ficheiro de dados gerado pelo instrumento para a plataforma O-Pitblast® e verificar o desvio da perfuração em relação ao planeado: no menu *Free-Face* está presente uma secção relativa ao desvio de furos, onde poderá ser importado o ficheiro de dados do Boretrak. Na figura Figura 67 – Exemplo de furo real desviado é visível um exemplo com um furo teórico (a azul) e o respetivo furo real com registo do Boretrak (a vermelho), que permite concluir que, sensivelmente a metade do seu comprimento o furo sofre um ligeiro desvio não só ao longo do seu comprimento, mas também para a esquerda do furo teórico.

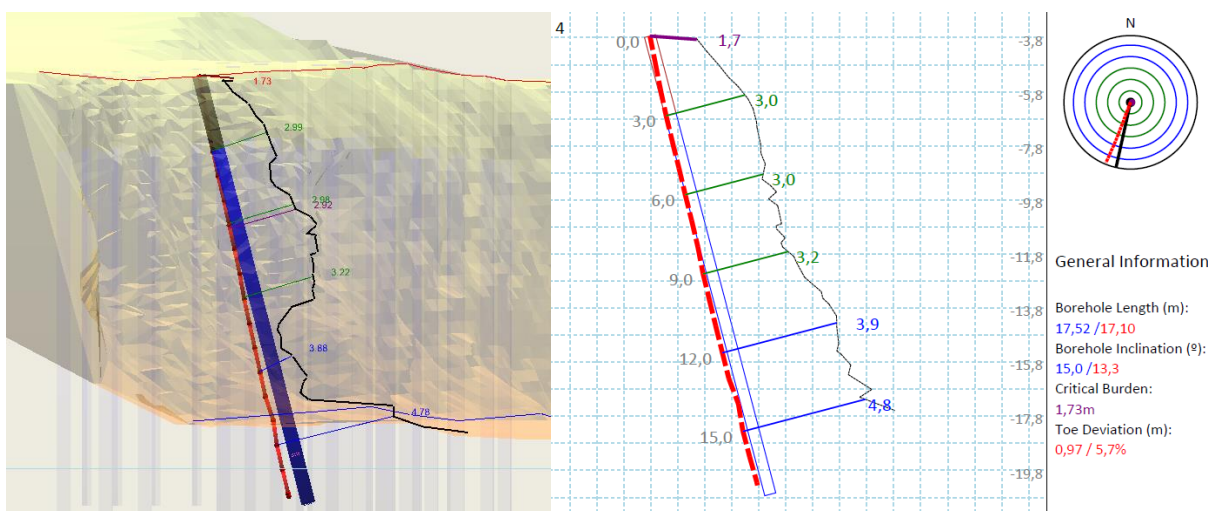


Figura 67 – Exemplo de furo real desviado relativamente ao teórico, com utilização de Boretrak

Mais uma vez, poderá ser necessário ajustar alguns parâmetros da pega de fogo, compensando estes desvios e tentando fazer com que o desmonte decorra sem qualquer tipo de perigo ou de maneira ineficaz.

5.2.5. ANÁLISE DO DESMONTE

Durante o desmonte, deverá ser feita uma gravação com recurso a câmara de alta velocidade, de modo a verificar se existem projeções de rocha assinaláveis, e analisar o comportamento do material rochoso durante as fases de rotura e fragmentação nas duas metades da pega de fogo.

Após o desmonte, com a obtenção do material fragmentado, é necessário captar algumas imagens de ambas as pilhas e, com recurso a um *software* de fotoanálise como o Wipfrag, obter as curvas granulométricas correspondentes a cada pilha.

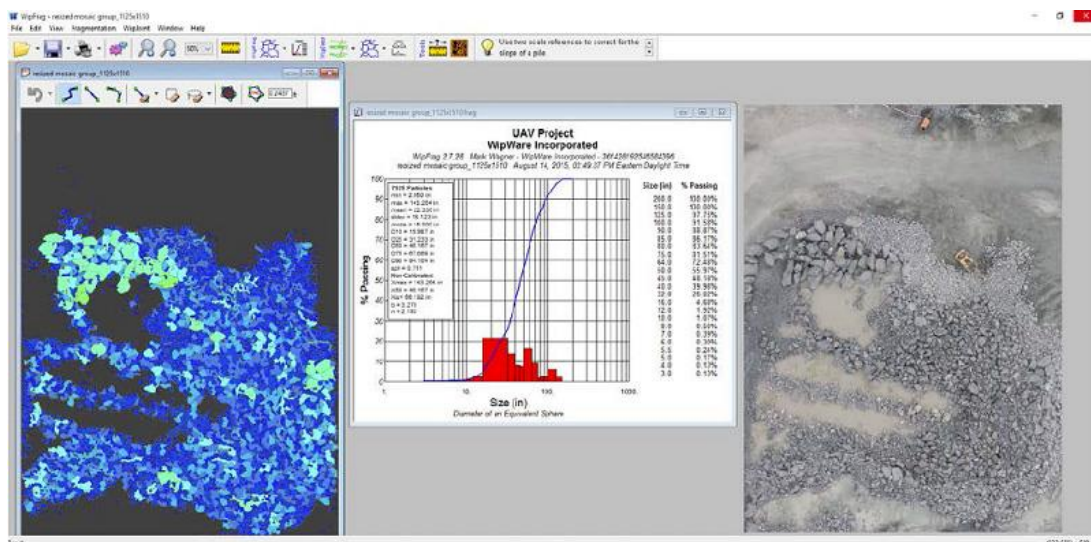


Figura 68 – Análise de fragmentação com Wipfrag
(<https://pix4d.com/supporting-blasting-operations-with-drones/>)

Através da análise com o Wipfrag, é possível verificar a distribuição granulométrica dos produtos obtidos no desmonte, e concluir se a fragmentação do material melhora com a utilização do algoritmo proposto neste trabalho. A melhoria na fragmentação deverá refletir-se sobretudo numa melhor distribuição granulométrica do material nos calibres que se pretendem obter com o desmonte, reduzindo tanto a quantidade de blocos como de finos. Isto faz com que possam ser reduzidos (ou até nulos) os custos com o transporte de blocos e de fragmentação secundária (taqueio), uma vez que poderão ser obtidos produtos com calibre adequado à boca do britador primário.

O algoritmo deverá ser testado em vários desmontes, maciços de geometria e litologia variadas e com vários tipos de pegadas de fogo, sendo que a médio-longo prazo, poderá ser ainda realizada uma análise custo-benefício, de modo a perceber se é possível conciliar os benefícios técnicos da sua utilização com os custos inerentes às mudanças efetuadas no plano de fogo - nomeadamente custos de perfuração, que poderão aumentar ligeiramente com a diferença da inclinação dos furos, mas que poderão ser bastante atenuados com a redução de custos em fragmentação secundária.

6

CONCLUSÕES

6.1. CONSIDERAÇÕES FINAIS

Com a conclusão desta Dissertação, importa tecer algumas considerações sobre o trabalho desenvolvido.

A metodologia aqui apresentada, com recurso a conceitos fundamentais de desmonte de rocha, programação linear, álgebra linear e geometria analítica, permite mais uma vez demonstrar a multidisciplinaridade da engenharia, em particular da engenharia de minas, e a importância que as várias ferramentas matemáticas possuem no auxílio à resolução de problemas diários nesta área.

Com a utilização desta ferramenta, prevê-se uma melhoria significativa na fragmentação obtida e um decréscimo na geração de vibrações e projeções de rocha, uma vez que a distribuição de cargas em relação à geometria da frente livre será bastante mais uniforme, em comparação com o dimensionamento-padrão de furos sem recurso ao algoritmo.

Apesar de não ter sido possível concretizar a validação do modelo durante o período temporal da realização deste trabalho, foi possível verificar o funcionamento correto do algoritmo em pelo menos dois casos práticos, não tendo sido detetados quaisquer erros ou anomalias nos resultados obtidos. Importa ressaltar que os efeitos deste algoritmo de otimização numa execução prática dependerão sempre de alguns fatores mais ou menos controláveis, como é exemplo a variabilidade do maciço a desmontar, o dimensionamento da pega de fogo, a qualidade dos equipamentos de perfuração e a execução técnica do perfurista.

Para além disso, esta ferramenta permitirá poupar algum tempo no dimensionamento de pegadas de fogo, uma vez que basta a aquisição de dados da frente livre e dos furos para obter um cálculo automático dos parâmetros de inclinação e de afastamento de cada um dos furos. Uma vez incorporada num *software* como o O-Pitblast®, a sua eficácia será ainda mais notória, uma vez que se dispensam as etapas de importação de ficheiros para MATLAB, e exportação dos dados otimizados, fazendo com que este processo seja praticamente instantâneo. Numa indústria cuja prioridade é maximizar as condições de segurança das operações extrativas, e em simultâneo rentabilizar a produção, este algoritmo poderá, de facto, ser uma enorme mais-valia para muitas empresas.

6.2. ALGUMAS SUGESTÕES PARA TRABALHOS FUTUROS

Uma vez testada a eficácia do algoritmo proposto nesta Dissertação, o mesmo poderá ser incorporado a curto-prazo numa plataforma de modelação e dimensionamento de desmontes a céu aberto com recurso a explosivos, como é o caso do O-Pitblast®.

A validação do modelo aqui apresentado é de extrema importância para a confirmação dos seus efeitos previstos. Aconselha-se a sua experimentação em vários tipos de litologia e condições geotécnicas, tendo sempre em atenção a correta execução dos furos e potenciais desvios, e os restantes parâmetros da pega de fogo.

Dado que este algoritmo representa apenas um processo de otimização da inclinação dos furos de desmonte, futuramente poderão ser estudadas hipóteses para a otimização de outros parâmetros geométricos em pegas de fogo, de modo a complementar de modo otimizado todo o processo de dimensionamento de um desmonte.

REFERÊNCIAS BIBLIOGRÁFICAS

- Albers, D. R. C. & D. G., 1986. An Interview with George B. Dantzig: The Father of Linear Programming. *The College Mathematics Journal*, Vol. 17, No. 4 (Sep., 1986), pp. 292-314.
- Aziz, N., 2012. Detection of Outliers in Multivariate Data: A Method Based on Influence Eigen. *Proceedings of the World Congress on Engineering 2012*, Vol 1.
- Bhandari, S., 1997. *Engineering Rock Blasting Operations*. Rotterdam: Balkema.
- Boulos, P., 1987. *Geometria Analítica: um tratamento vetorial*. São Paulo: Makron Books.
- Cardu, M., Giraudi, A. & Oreste, P., 2013. A review of the benefits of electronic detonators. *Revista Escola de Minas, Ouro Preto*, 66(3), pp. 375-382.
- Domingo, J., Leite, F., Miranda, V. & Carrasco, I., 2015. Dilution, Ore Grade and Blast Movement Calculation Model. *Fragblast 11, 11th International Symposium on Rock Fragmentation by Blasting, Sydney*, pp. 71-78.
- Dyno Nobel, 2011. *Product Application Manual: NONEL® and Primacord®*. Salt Lake City: s.n.
- Gama, J., 2010. *Extração de Conhecimento de Dados: Data Mining*. Lisboa: Edições Sílabo.
- Gass, S. & Assad, A., 2005. *An Annotated Timeline of Operations Research: An Informal History*. Berlin: Springer-Verlag.
- Gujarati, D., 1978. *Basic Econometrics*. New York: McGraw Hill.
- Gustafsson, R., 1981. *Blasting Technique*. Vienna: Dynamit Nobel Wien Gesellschaft m.b.H..
- Hillier, F., 2010. *Introduction to Operations Research*. New York: McGraw Hill.
- Hustrulid, W., 1999. *Blasting Principles for Open Pit Mining*. Rotterdam: Balkema.
- Jimeno, E. L., Jimeno, C. L. & Bermudez, P. G., 2003. *Manual de Perforación y Voladura de Rocas*. Madrid: Instituto Geológico y Minero de España.
- Kaya, A., Bulut, F. & Alemdag, S., 2011. Applicability of excavability classification systems in underground excavations: A case study. *Scientific Research and Essays Vol. 6(25)*, pp. 5331-5341.
- Lipschutz, S., 1991. *Álgebra Linear: teoria e problemas*. São Paulo: Makron Books.
- Lownds, C., 1986. The strength of explosives. *The planning and operation of open-pit and strip mines*, pp. 151-159.
- McKenzie, A., 1966. Cost of explosives—do you evaluate it properly?. *Mining Congress Journal*, pp. 32-41.
- Miranda, V., Couceiro, P. & Blazquez, J., 2014. *Rioblast: Software de diseño y simulación*. Madrid, s.n.
- Miranda, V. G., 2016. *Validação de Modelos: Uma Análise Residual*, Porto: FEUP.
- Murteira, B., Andrade e Silva, J., Pimenta, C. & Silva Ribeiro, C., 2008. *Introdução à Estatística (2ª Edição)*. Lisboa: McGraw Hill.
- Ryan, P. J., 1986. *Euclidean and non-Euclidean geometry: and analytical approach*. Cambridge: Cambridge University Press.
- Sanchidrián, J. A. & Muñiz, E., 2000. *Curso de Tecnología de Explosivos*. Madrid: Servicio de Publicaciones de la Fundación Gómez-Pardo.

Taha, H., 2008. *Pesquisa Operacional: Uma Visão Geral*. São Paulo: Pearson Prentice Hall.

Tormos Juan, P. & Lova Ruiz, A., 2003. *Investigacion Operativa para Ingenieros*. Valencia: Universidade de Valencia.

Union Española de Explosivos, 1997. *Manual de Empleo de Explosivos*. Madrid: Gráficas Arias Montano, S.A..

<http://www.detnet.com/products/underground-mining/digishot-plus>

https://www.dynonobel.com/~media/Files/Dyno/ResourceHub/Guides/NONEL_PrimacordAppManual/41160%20NonelPrimacord%20Manual.pdf

<http://www.kayakujapan.co.jp/en/sale/denkiraikan.html>

<https://www.maxam.net/>

<http://mathworld.wolfram.com/Covariance.html>

http://www.oricaminingservices.com/ee/en/product/products_and_services/initiating_systems/page_initiating_systems/electric_detonator_sd/1385

<https://optron.com/carlson/products/rodded-boretrak/>

<https://pix4d.com/supporting-blasting-operations-with-drones/>

<http://radiology.rsna.org/content/217/3/897/F1.expansion.html>

<http://resources.renishaw.com/es/details/folleto-boretrak-con-barras-y-boretrak-con-cable--66251>

ANEXOS

ANEXO A

ANEXO A.1. – SCRIPT “PRINCIPAL”

```
%principal

%limpar os graficos
delete(findall(0,'Type','figure'));

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%limpar as variaveis
clear all;
clc;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Leitura dos dados

data=csvread('terrain.csv',1,0); %dados de terreno
collar=csvread('holes.csv',1,0); %dados dos furos
toe = collar(:,4:6); % colunas relativas aos pés dos furos
collar = collar(:,1:3); %colunas relativas às cristas dos furos

% ATENÇÃO -> o arquivo pode ser substituído para novo terreno ou novos
% furos (basta mudar nome dos ficheiros)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Input de variáveis

maxPoints=1000; % valor máximo de pontos para a redução de amostra
stemming = 3; % valor do tampão
maxBurden = 3; % valor do afastamento crítico

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%preparar os dados
x=data(:,1);y=data(:,2);z=data(:,3);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% grafico terreno todos pontos
scatter3(x,y,z,2,'filled');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%novo grafico
figure(2);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%grafico com furo
scatter3(x,y,z,2,'filled');
hold on;
[m n] = size(collar);
for i=1:m
    line([collar(i,1) toe(i,1)], [collar(i,2) toe(i,2)], [collar(i,3)
toe(i,3)],...
        'LineWidth', 5, 'Color', 'black')
end
```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%amostra menor para ganhar tempo de processamento
[m n ] = size(data);
randomic = randi(m,maxPoints+1000,1);

% a função randi destina-se a obter uma amostra aleatória dos pontos que
% constam na nuvem inicial

randomic = unique(randomic);
[m n ] = size(randomic);

if m>maxPoints
    randomic = randomic(1:maxPoints);
end

temp=[];

[maxPoints n] = size(randomic);

for i=1:maxPoints
    temp=[temp;data(randomic(i),:)];
end

data=temp;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%novo grafico com amostra menor

figure(3);
x=data(:,1);y=data(:,2);z=data(:,3);
scatter3(x,y,z,2,'filled');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%line([5062.26 5062.26],[7410.32 7410.32], [-69.48 -91.06], 'LineWidth', 5,
'Color', 'black')

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%posicao do furo, pe, tampao e discretizacao dos pontos ao longo do furo

[m n] = size(collar);

result=[];

figureNumber=4;

for hole =1:m

    %Gráfico com terreno de amostra reduzida e fila de furos verticais

    figure(figureNumber);
    figureNumber=figureNumber+1;

    scatter3(x,y,z,2,'filled');
    hold on;

    holePoints=[];

```

```

director = toe(hole,:) - collar(hole,:); % vetor diretor do furo
holeLength = norm(director); % comprimento do furo
step = holeLength/40; % determinar o passo de discretização
director=director/holeLength;
stemmingPosition=collar(hole,:)+director*stemming; %posição do tampão

for i=1:50

    holePoints=[holePoints;collar(hole,:) + director*step*i];
    % calcula os pontos discretizados ao longo do furo, segundo um
    % passo de discretização, e para os sucessivos índices de 1 até 50

end

lastToe = holePoints(50,:); % posição do ponto do pé do furo

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Pontos extra:
% são usados para fazer o prolongamento dos pontos do furo
relativamente a
% frente livre, de modo a que possam ser encontrados pontos-candidatos
% adequados aos pontos da zona do pé do furo (preparação para a
% obtenção do perfil crítico: adição de 10 pontos

maxDist=100000;
index = 0;

for j=1:maxPoints

    M=[lastToe; data(j,:)];

    dist = pdist(M,'euclidean');
    if dist<maxDist
        maxDist=dist;
        index = j;
    end
end

newDirection = data(index,:) - collar(hole,:);
newDirection = [newDirection(1) newDirection(2) 0];
auxLength = 10;
step = 0.5;
newDirection=newDirection/auxLength;

for i=1:10

    holePoints=[holePoints;lastToe + newDirection*step*i];

end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%afastamento critico
[m n] = size(holePoints);
criticalBurden=[];

```

[illegible]


```

line(criticalBurden(:,1),criticalBurden(:,2),criticalBurden(:,3),'Color','red');

scatter3(criticalBurden(:,1),criticalBurden(:,2),criticalBurden(:,3),15,'filled','red');
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Eliminar outliers

criticalBurden = eraseOutliers( criticalBurden, collar(hole,:),
toe(hole,:) );

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Procedimento para encontrar uma base nova

%eliminar dados duplicados

criticalBurdenVariance = unique(criticalBurden, 'rows');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Covariância

covar = cov([criticalBurdenVariance(:,1),...
criticalBurdenVariance(:,2),criticalBurdenVariance(:,3)]);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Cálculo dos valores e vetores próprios

[Ave AVa] = eig(covar);

Vector = [];

% Encontrar maior valor próprio e vetor próprio associado

if AVa(3,3)>AVa(2,2) && AVa(3,3)>AVa(1,1)

    Vector=Ave(:,3);

else

    if AVa(2,2)>AVa(1,1)
        Vector=Ave(:,2);
    else
        Vector=Ave(:,1);
    end

end

% Base nova, formada pelo vetor próprio associado ao maior valor

baseX = [Vector(1) Vector(2) 0];
baseX=baseX/norm(baseX);
baseZ=[0 0 1];
baseY = cross(baseX,baseZ);
baseY=baseY/norm(baseY);

```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Mudança de base (ver função getPointsNewBase.m)

criticalBurden = getPointsNewBase(-baseZ, baseX,
baseY,criticalBurden,collar(hole,:), toe(hole,:));

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Transladar

minX=min(criticalBurden(:,1));
minY=min(criticalBurden(:,2));
minZ=min(criticalBurden(:,3));

[m n] = size(criticalBurden);

for i=1:m
    criticalBurden(i,:) = [criticalBurden(i,1) - minX ...
        criticalBurden(i,2) - minY criticalBurden(i,3) - minZ];
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%eliminar dados duplicados
criticalBurden = unique(criticalBurden, 'rows');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%pontos preparados em 2D

figure(figureNumber);
figureNumber=figureNumber+1;

line(criticalBurden(:,1),criticalBurden(:,2))

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

dataSolver=[];

[m n] = size(criticalBurden);

% considerar apenas os pontos para lá do tampão

for i=1:m

    if (criticalBurden(i,1)>stemming)

        dataSolver=[dataSolver;criticalBurden(i,:)];
    end
end

[m n] = size(dataSolver);

m=m-1; %(n° de restrições)

% construção das matrizes A e b

A=zeros (m*4+1,m+2);

```

```

B=zeros(m*4+1,1);
index=1;

for i=1:m
    A(index,2)= dataSolver(i+1,1)-dataSolver(i,1);
    A(index+1,2)= dataSolver(i,1)-dataSolver(i+1,1);
    A(index,i+2)= -1;
    A(index+1,i+2)= -1;

    B(index,1)=dataSolver(i+1,2)-dataSolver(i,2);
    B(index+1,1)=dataSolver(i,2)-dataSolver(i+1,2);

    index=index+2;
end

for i=1:m
    A(index,2+i)= -1;

    B(index,1)=0;

    index=index+1;
end

for i=1:m+1
    A(index,1)= 1;
    A(index,2)= dataSolver(i,1);

    B(index,1)=-maxBurden+dataSolver(i,2);

    index=index+1;
end

% definir soma de Xi, para multiplicação por B
B=B';

somaX=0;

for i=1:m+1
    somaX=somaX+dataSolver(i,1);
end

%função objetivo
f=zeros(1,m+2);

f(1,1)=-(m+1);
f(1,2)=-somaX;

for i=1:m
    f(1,i+2)=1;
end

% resolução do problema com linprog
[solution,fval,exitflag,output] = linprog(f,A,B);
clc;

% matriz de resultados

```

```

        result=[result;[hole , criticalBurden(1,1)-
solution(1)+solution(2)*criticalBurden(1,1) , ...
atan(solution(2))/pi*180]];

        hold on
        line(criticalBurden(:,1),criticalBurden(:,1)*solution(2)+solution(1),
'Color', 'red')
end

result

Hole_Number=result(:,1);
Burden=result(:,2);
Inclination=result(:,3);
resultado=table(Hole_Number,Burden,Inclination);

resultado

```

ANEXO A.2. – FUNÇÃO “PROJECTPOINTINLINE”

```
function projection = projectPointInLine(l1,l2,p)

% Esta função destina-se ao cálculo da projeção dos pontos da frente livre
% na reta que representa o furo

[m n] = size(p);

projection=[];

for i=1:m

    % vetor u, da crista ao ponto do perfil
    u = p(i,:)-l1;

    % vetor v, vetor diretor da reta que representa o furo
    v=l2-l1;
    director = l2-l1;
    director = director/norm(director);

    % Pontos projetados
    projection =[projection; l1 + director*dot(u,v)/norm(v)];

end

end
```


ANEXO A.3. – FUNÇÃO “ERASEOUTLIERS”

```
function result = eraseOutliers( criticalBurden, collar, toe )

% Esta função destina-se à detecção e exclusão de outliers presentes no
% perfil crítico

projection = projectPointInLine(collar,toe,criticalBurden);

distance = [];

[m n] = size(projection);

%cálculo das distâncias entre perfil crítico e projeção dos pontos

for i=1:m
    distance=[distance; ((criticalBurden(i,1)-projection(i,1))^2 + ...
        (criticalBurden(i,2)-projection(i,2))^2 + (criticalBurden(i,3)-
projection(i,3))^2)^0.5];
end

% ordenar dados

ordenedDistance = sort(distance);

% cálculo inter-quartil

q25 = quantile(ordenedDistance, 0.25);
q75 = quantile(ordenedDistance, 0.75);

difInterQuantile = q75-q25;

result=[];

for i=1:m

    if (distance(i)>q25-1.5*difInterQuantile &&
distance(i)<q75+1.5*difInterQuantile)

        result = [result;criticalBurden(i,:)];
        % output do perfil crítico sem outliers
    end

end

end
```


ANEXO A.4. – FUNÇÃO “GETPOINTSNEWBASE”

```
function points = getPointsNewBase(baseX,baseY,baseZ, points, collar, toe)

% Esta função destina-se ao cálculo dos pontos que formam a nova base

M=[baseX(1) baseY(1) baseZ(1); baseX(2) baseY(2) baseZ(2); baseX(3)
baseY(3) baseZ(3)];

% matriz de mudança de base

M=M^(-1);

[m n] = size(points);

for i=1:m

    %projeção dos pontos

    projection = projectPointInLine(collar,toe,points(i,:));

    % distância euclidiana entre furo e perfil

    dist = pdist([points(i,:); projection], 'euclidean');

    % mudança de base

    points(i,:) = (M*points(i,:)')';

    % coordenada y = distância entre furo e pontos

    points(i,2) = dist;

end

end
```


ANEXO B

Os dados presentes neste anexo foram recolhidos em ambiente experimental e são relativos a duas frentes livres distintas. Apresentam-se então dois casos práticos de execução do algoritmo, com recurso numa primeira fase à representação da bancada em O-Pitblast®, com furos modelados nesta plataforma. De seguida foi feita uma exportação dos dados para o algoritmo em MATLAB, e foram obtidas as tabelas de resultados. Posteriormente apresentam-se os furos modificados consoante os resultados da otimização.

Em ambas as situações foram considerados os seguintes valores de parâmetros de entrada:

- Número de pontos a utilizar na amostra = 2000;
- Valor de afastamento crítico = 3m;
- Valor de tamponamento = 3m.

ANEXO B.1. – CASO PRÁTICO Nº1

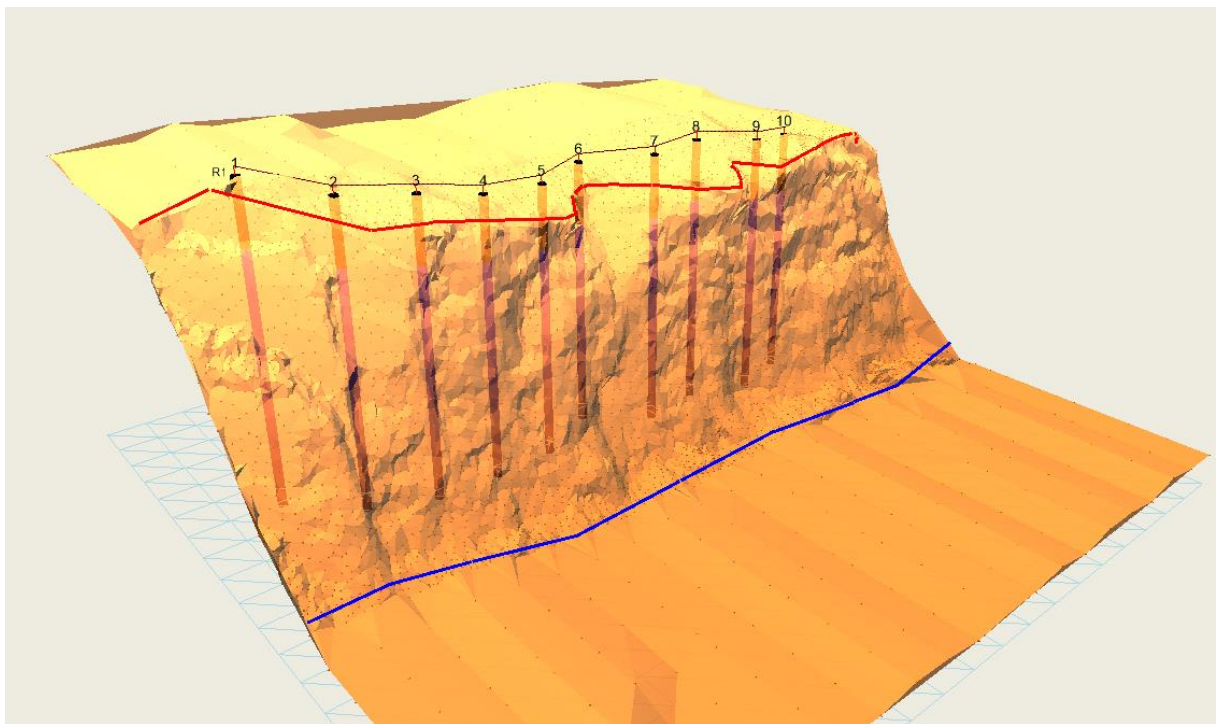


Figura B.1.1. – Bancada e furos modelados em O-Pitblast

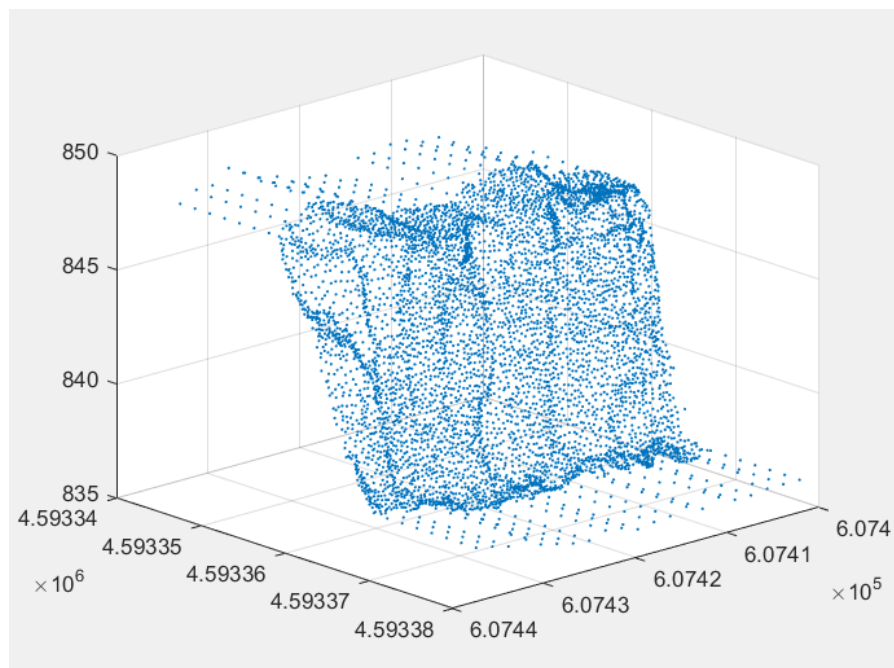


Figura B.1.2. – Nuvem de pontos da bancada em MATLAB

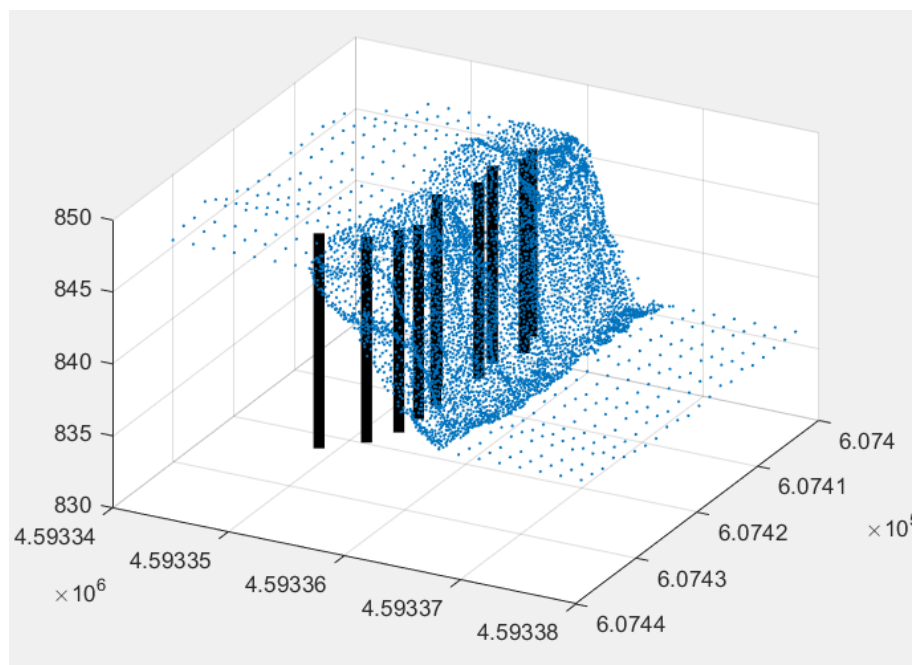


Figura B.1.3. – Nuvem de pontos da bancada e representação dos furos em MATLAB

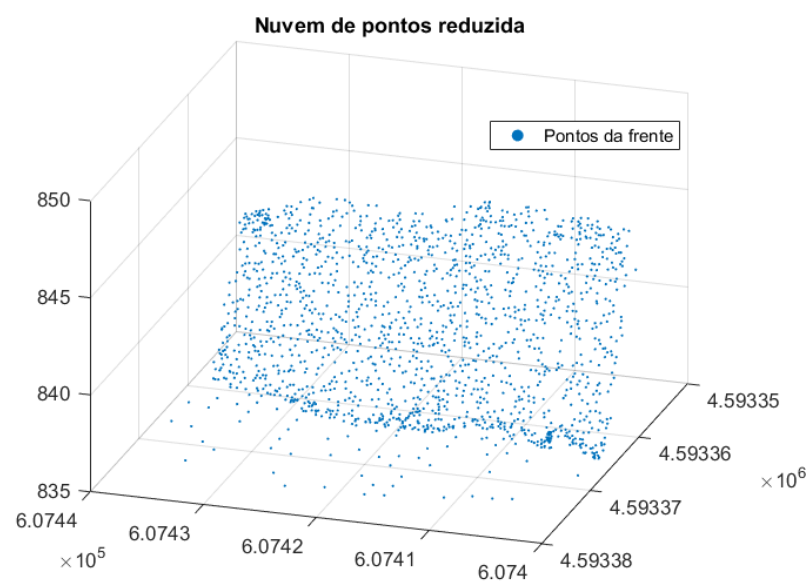


Figura B.1.4. – Amostra reduzida

Hole_Number	Burden	Inclination
1	3.5181	32.171
2	3.3781	27.392
3	4.0783	27.793
4	4.3543	28.617
5	4.2122	31.014
6	3.1973	26.115
7	2.8941	14.567
8	3.1005	18.861
9	4.0879	21.225
10	3.2362	22.022

Figura B.1.5. – *Output* de resultados

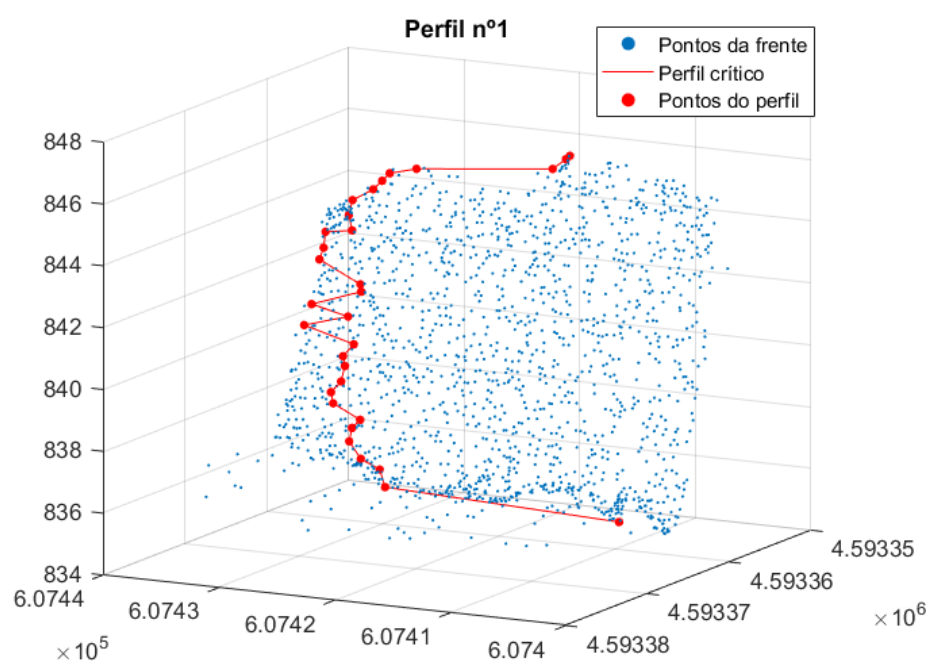


Figura B.1.5. – Perfil crítico nº1 (3D)

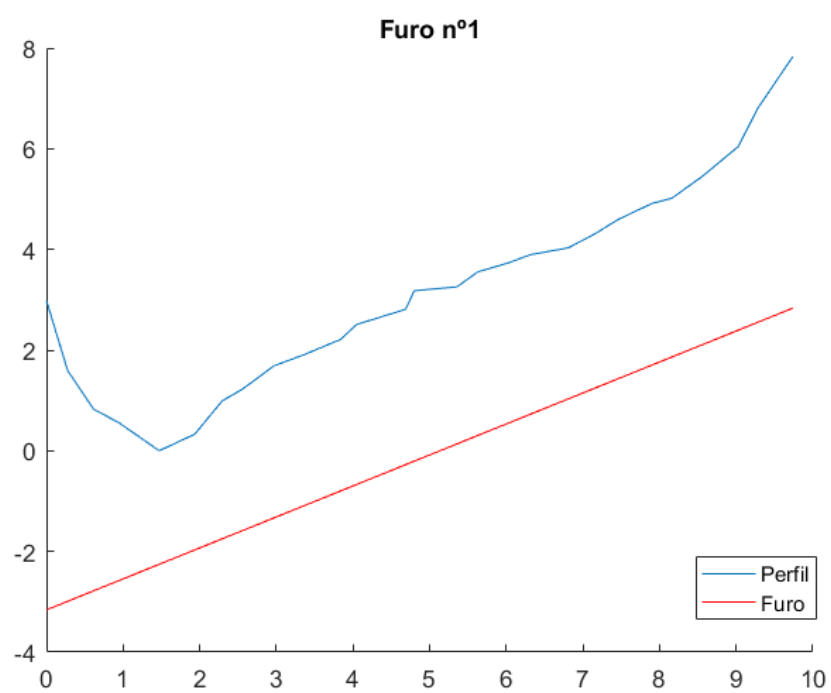


Figura B.1.6. – Furo nº1 (2D)

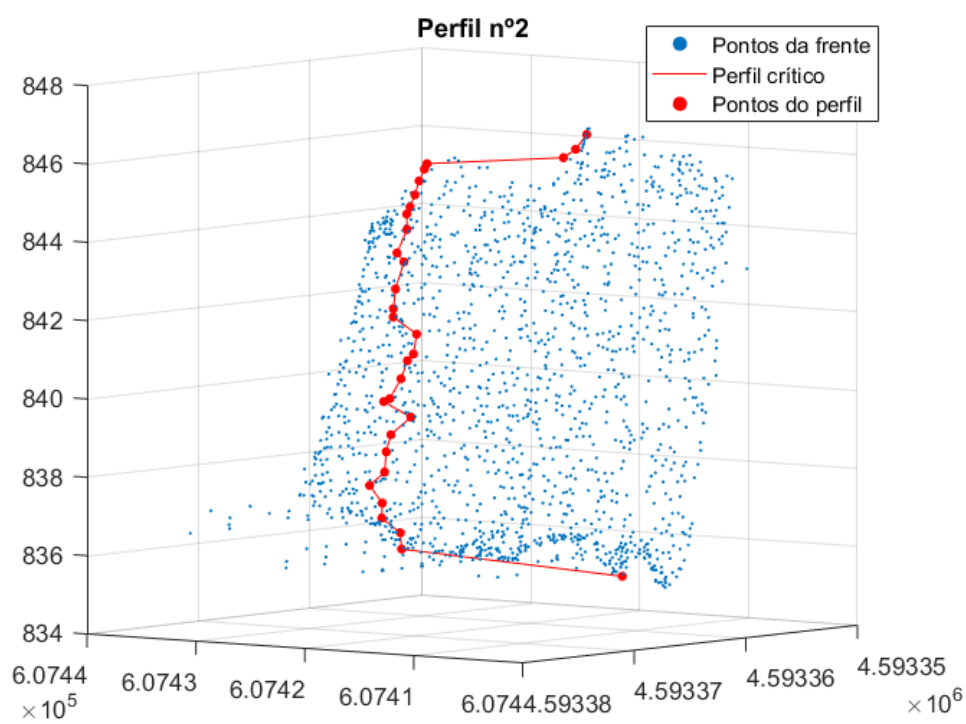


Figura B.1.7. – Perfil nº2 (3D)

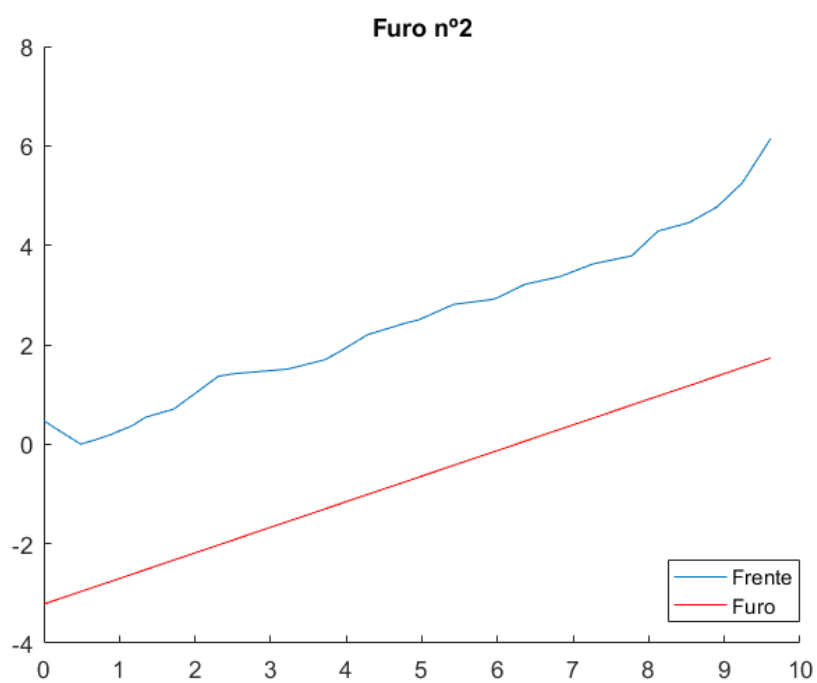


Figura B.1.8. – Furo nº2 (2D)

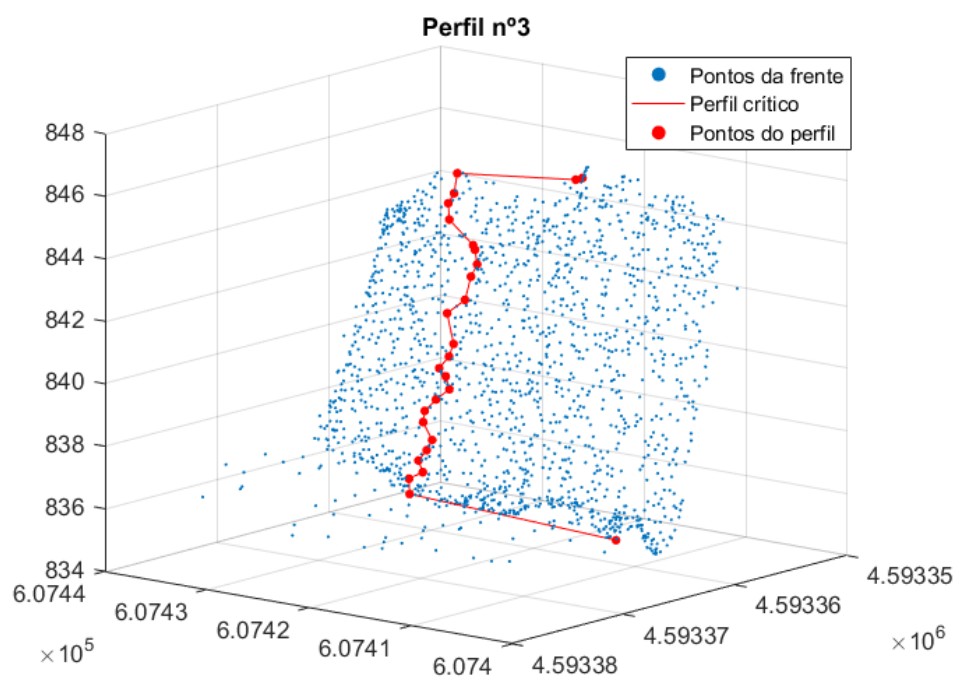


Figura B.1.9. – Perfil nº3 (3D)

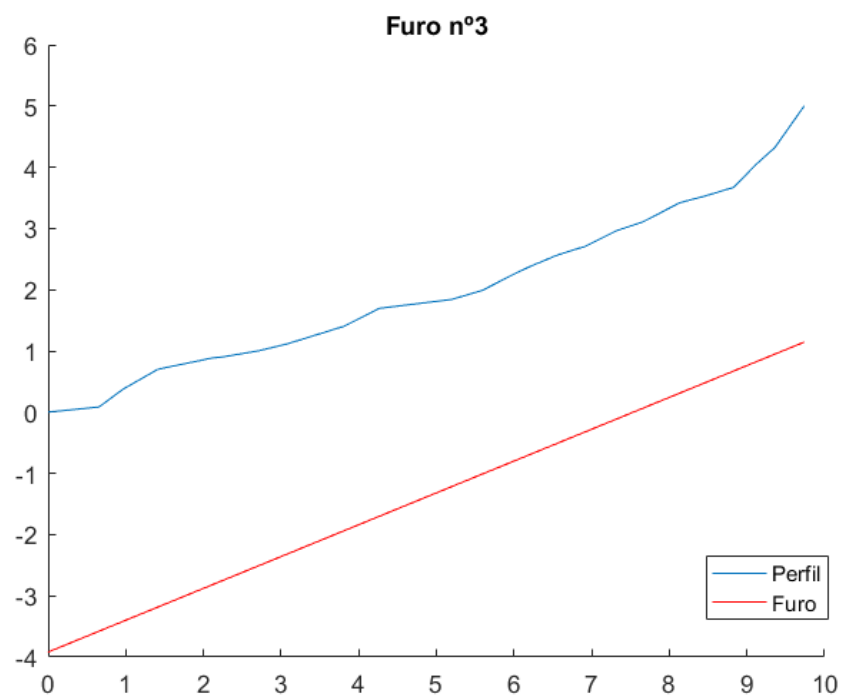


Figura B.1.10. – Furo nº3 (2D)

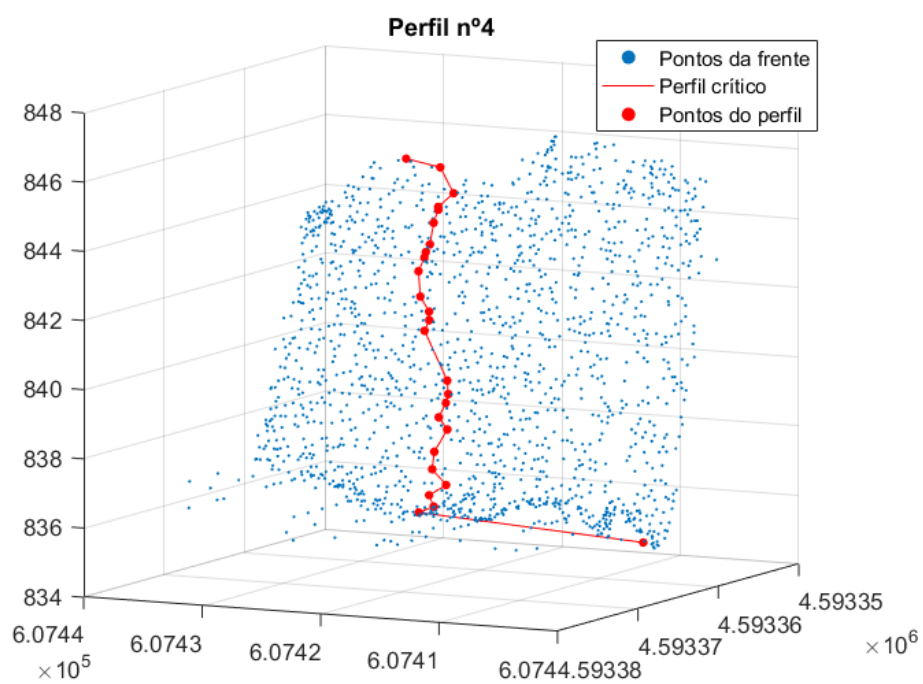


Figura B.1.11. – Perfil nº4 (3D)

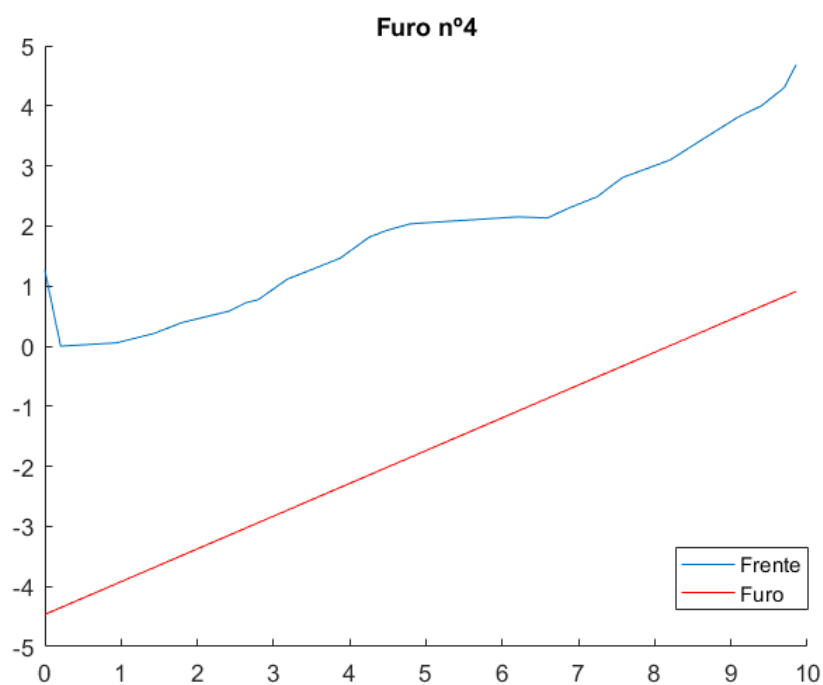


Figura B.1.12. – Furo nº4 (2D)

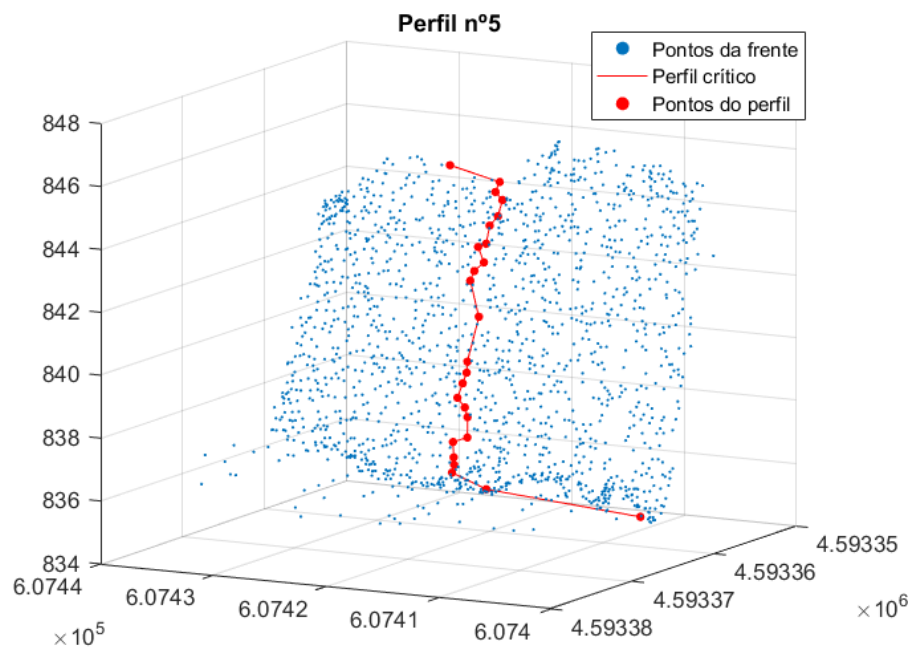


Figura B.1.13. – Perfil nº5 (3D)

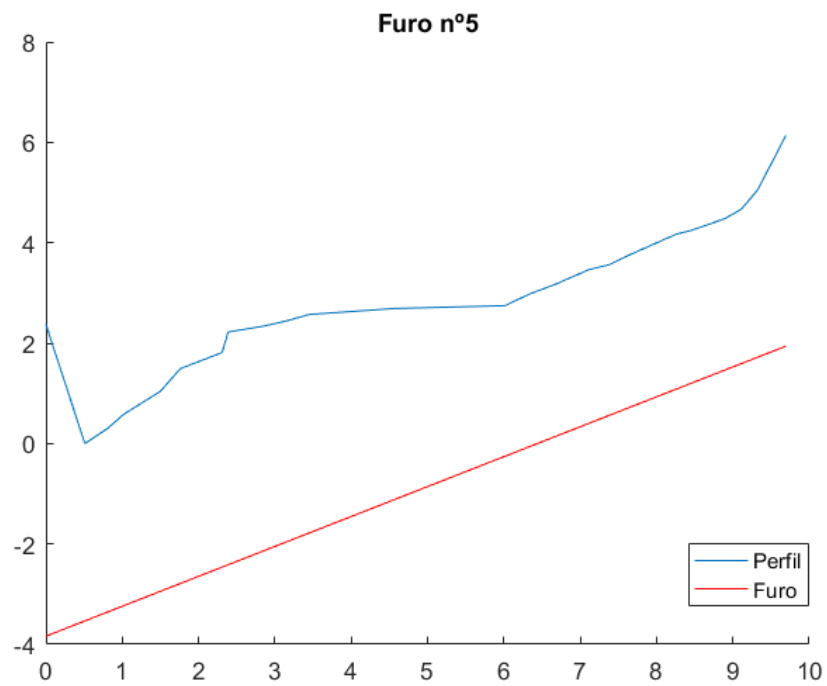


Figura B.1.14. – Furo nº5 (2D)

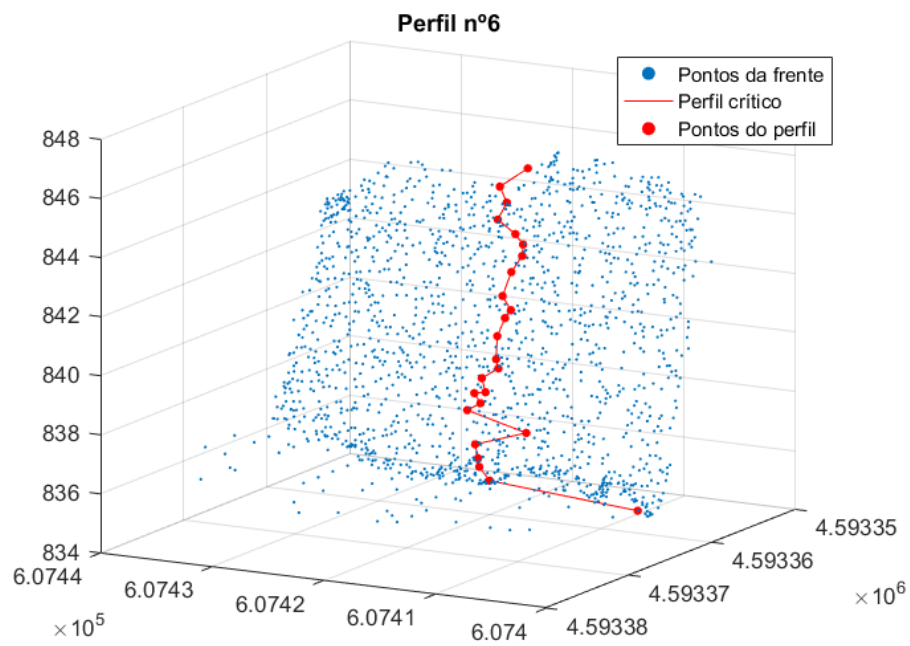


Figura B.1.15. – Perfil nº6 (3D)

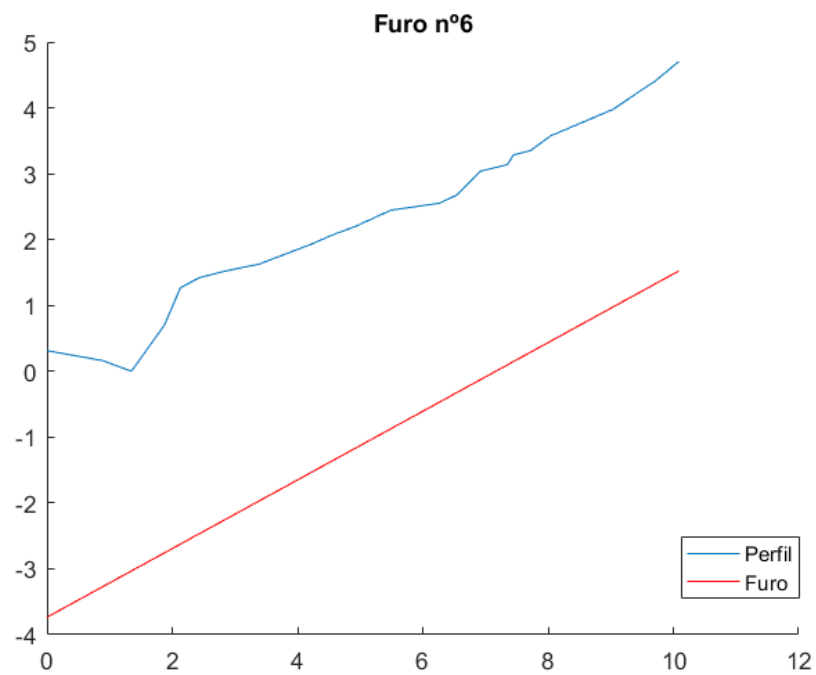


Figura B.1.16. – Furo nº6 (2D)

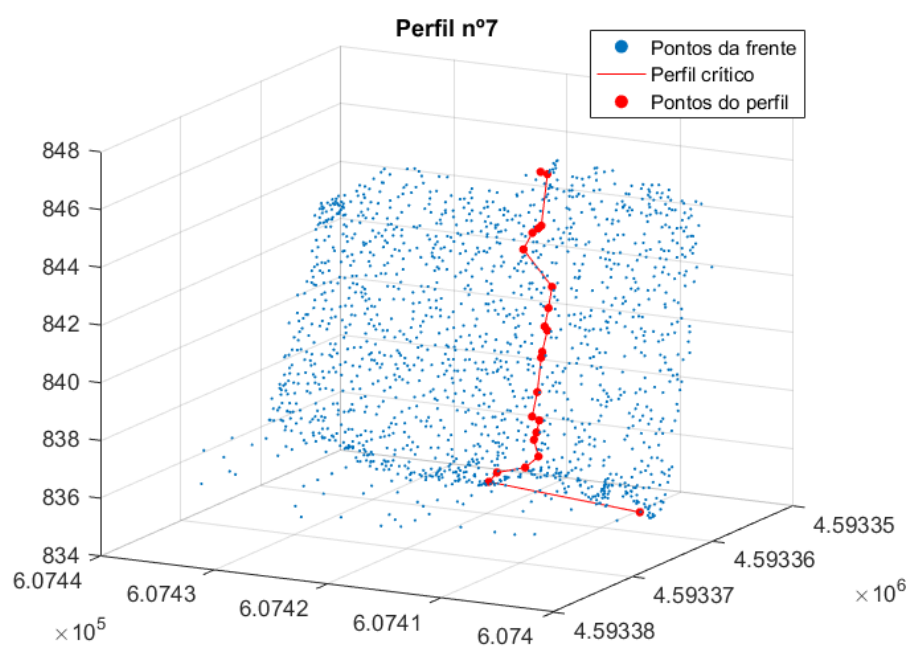


Figura B.1.17. – Perfil nº7 (3D)

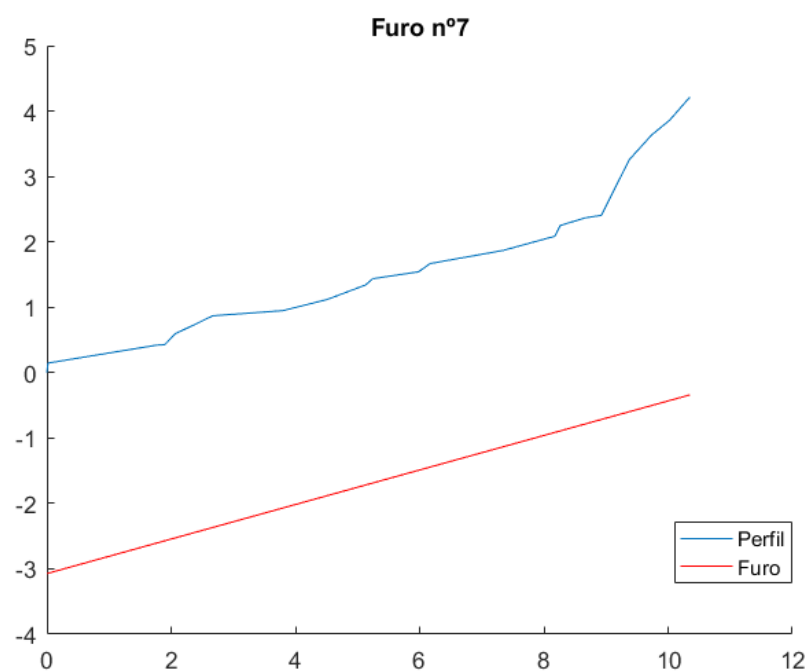


Figura B.1.18. – Furo nº7 (2D)

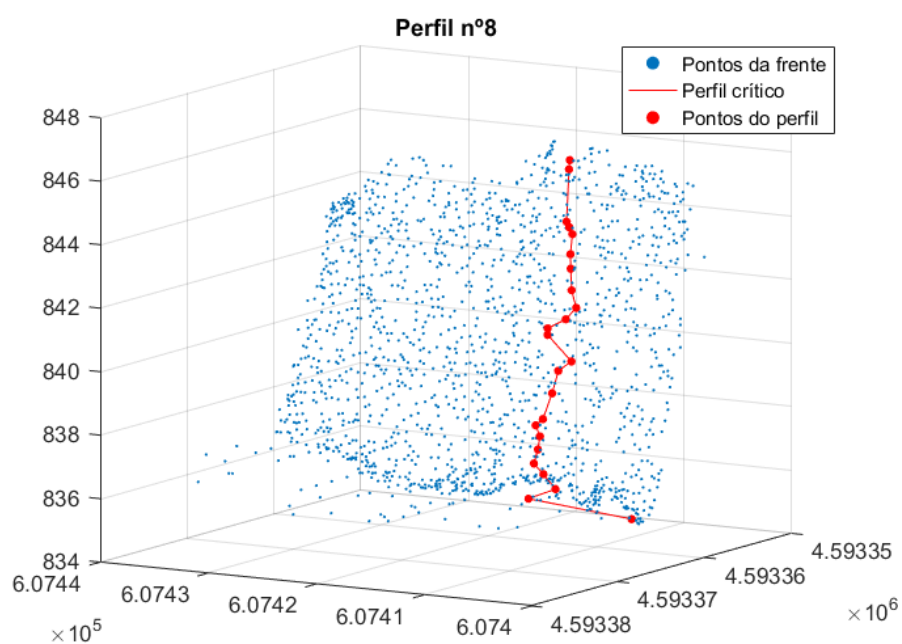


Figura B.1.19. – Perfil nº8 (3D)

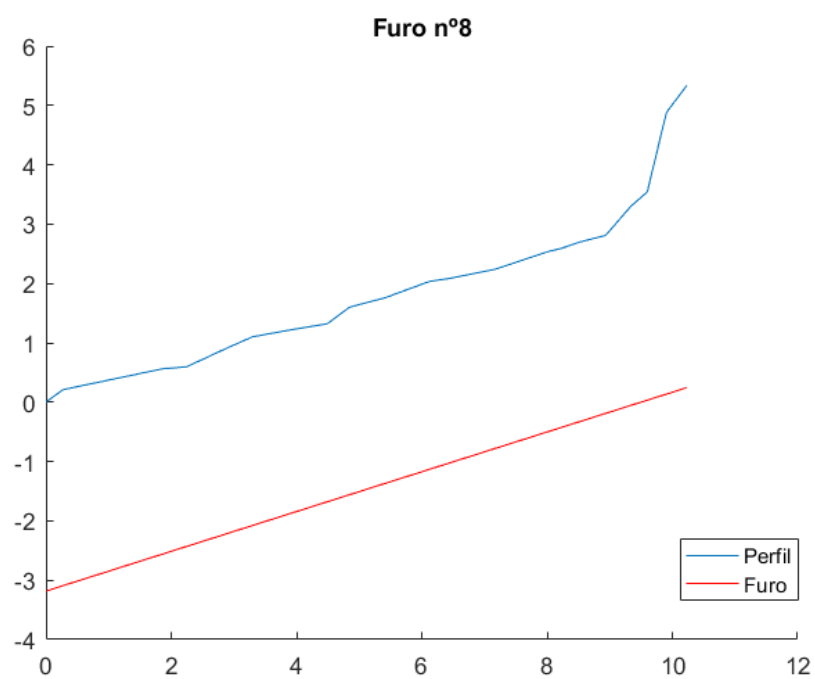


Figura B.1.20. – Furo nº9 (2D)

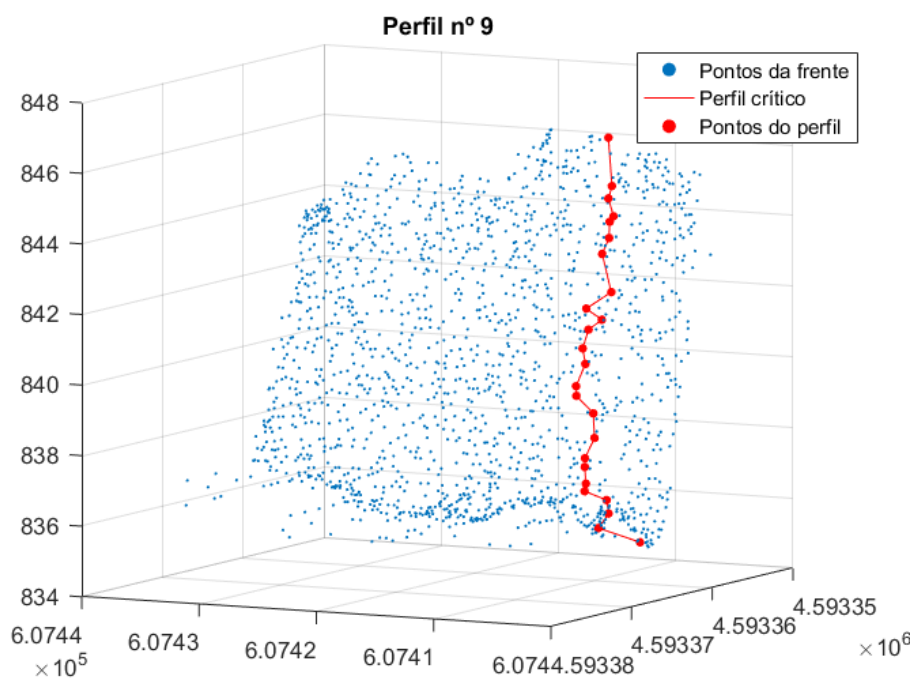


Figura B.1.21. – Perfil nº9 (3D)

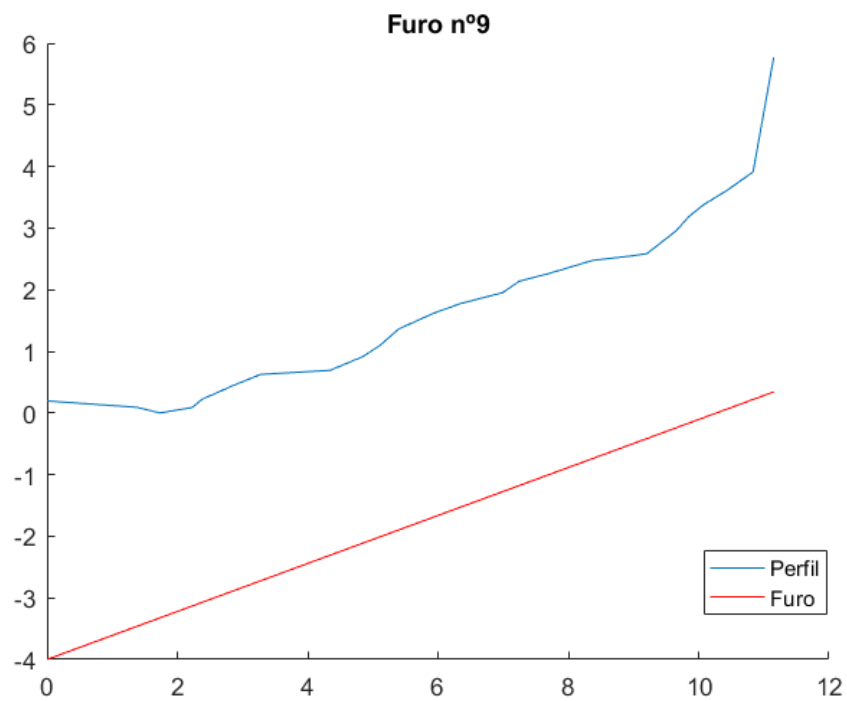


Figura B.1.22. – Furo nº9 (2D)

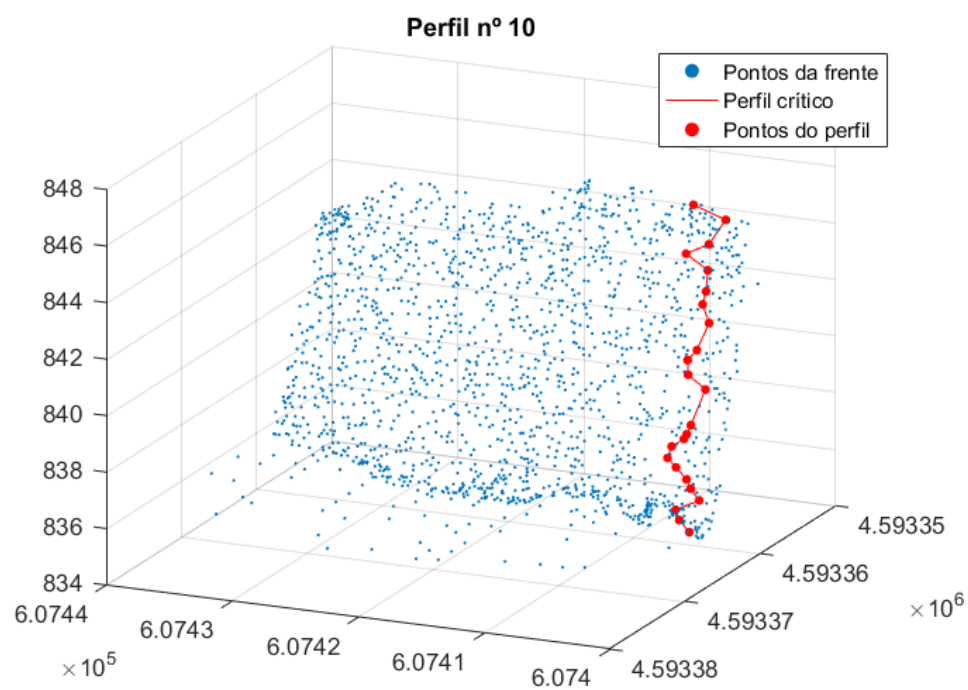


Figura B.1.23. – Perfil nº10 (3D)

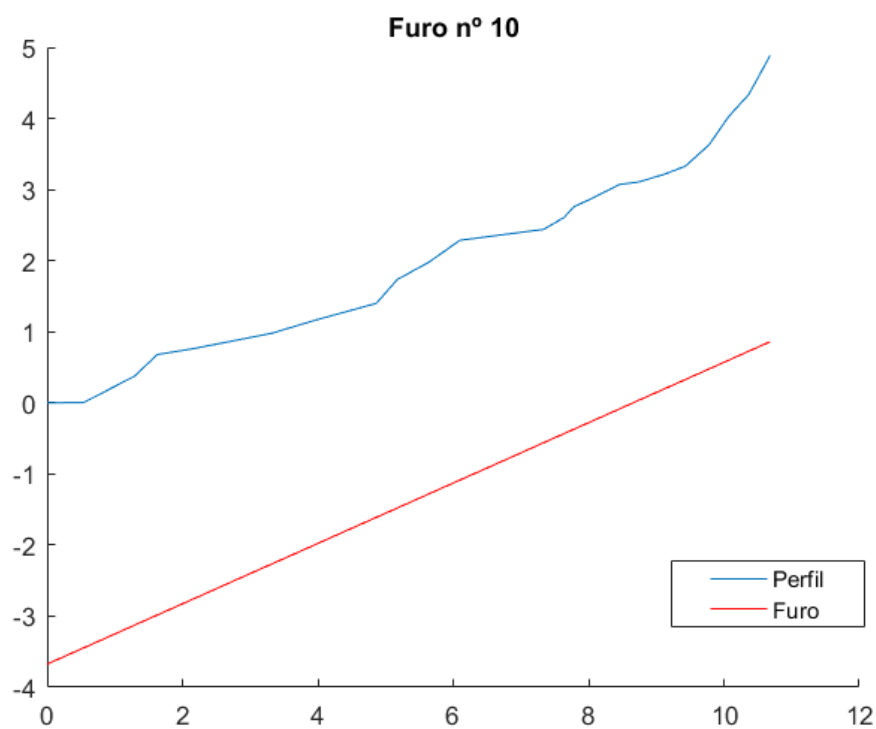


Figura B.1.24. – Furo nº10 (2D)

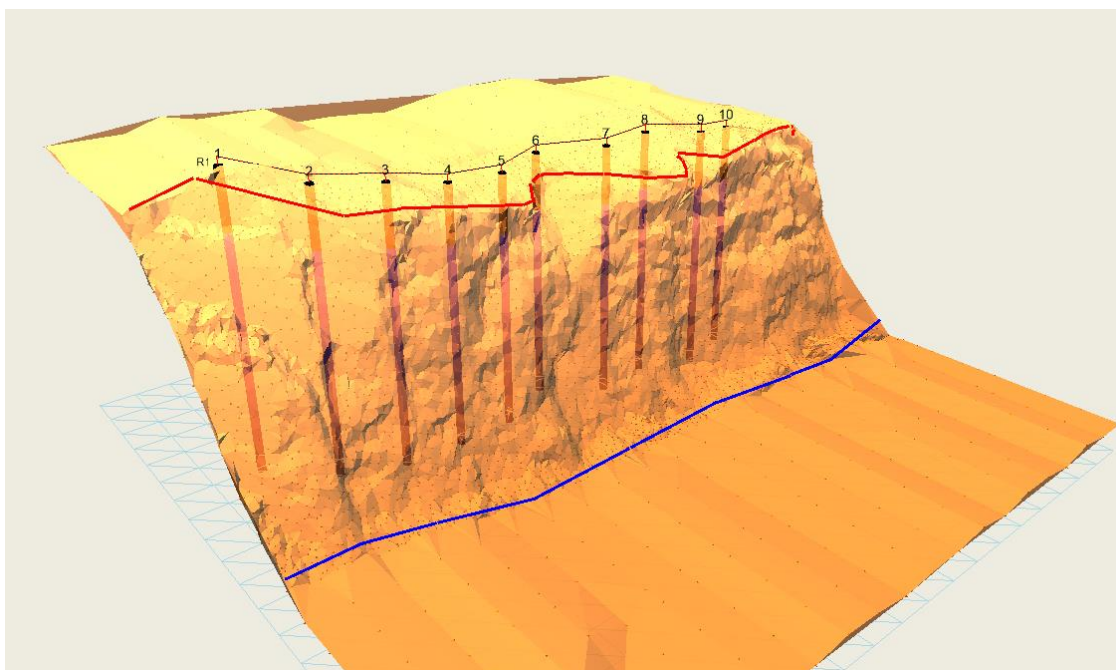


Figura B.1.25. – Bancada original com furos-padrão

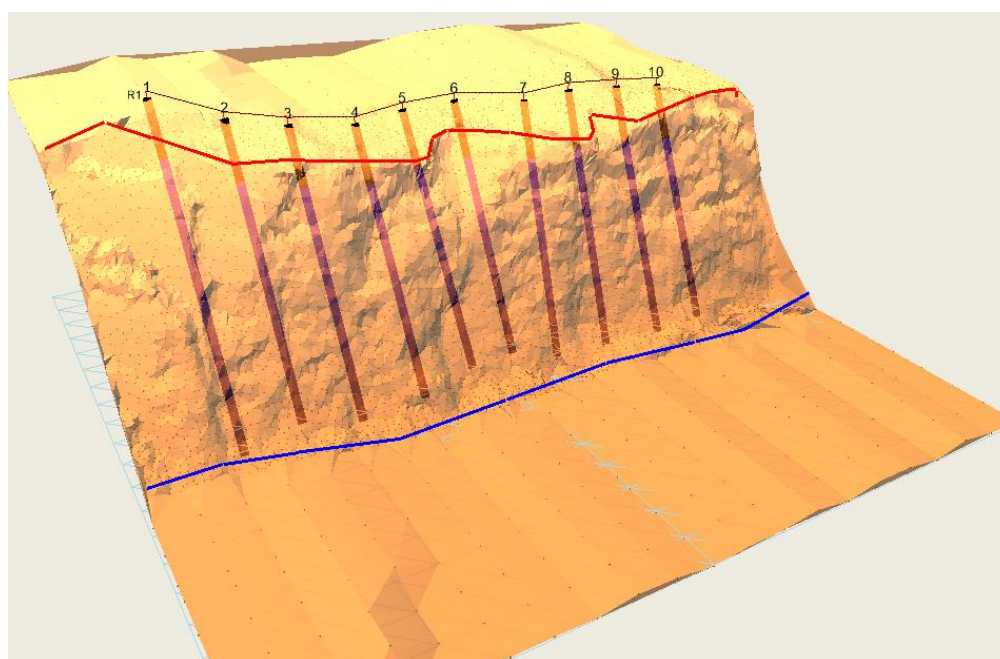


Figura B.1.26. – Importação dos furos otimizados para a plataforma O-Pitblast®

Comparando o resultado da figura B.1.26. com a figura B.1.25. é notória a diferença entre as inclinações dos furos após a otimização. É, contudo, necessário proceder a ajustes no comprimento dos furos após o processo de otimização, uma vez que os furos originais tinham um dado comprimento, que com a inclinação do furo poderá já não atingir o pé da bancada. O espaçamento também poderá ter de ser corrigido, de modo a que com os novos valores de afastamento, a malha permaneça o melhor espaçada possível.

ANEXO B.2. – CASO PRÁTICO Nº2

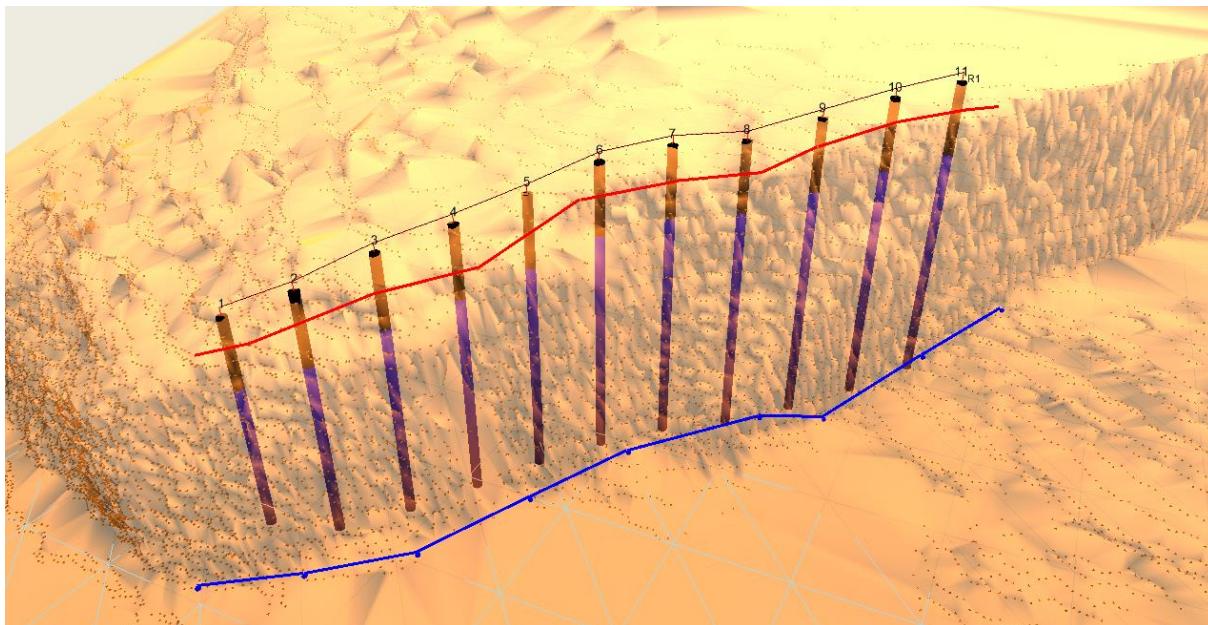


Figura B.2.1. – Modelo da bancada e furos

Após importar os dados da frente e dos furos para o MATLAB, executou-se o algoritmo, obtendo-se a seguinte tabela de valores:

Hole_Number	Burden	Inclination
1	2.9932	21.991
2	0.92992	4.9418
3	2.7387	2.0157
4	2.6647	6.8772
5	2.842	4.3138
6	2.8998	9.5405
7	3.7851	14.595
8	2.9455	14.123
9	2.0575	8.1612
10	3.205	8.2235
11	2.7743	7.3142

Figura B.2.2. – Output de parâmetros

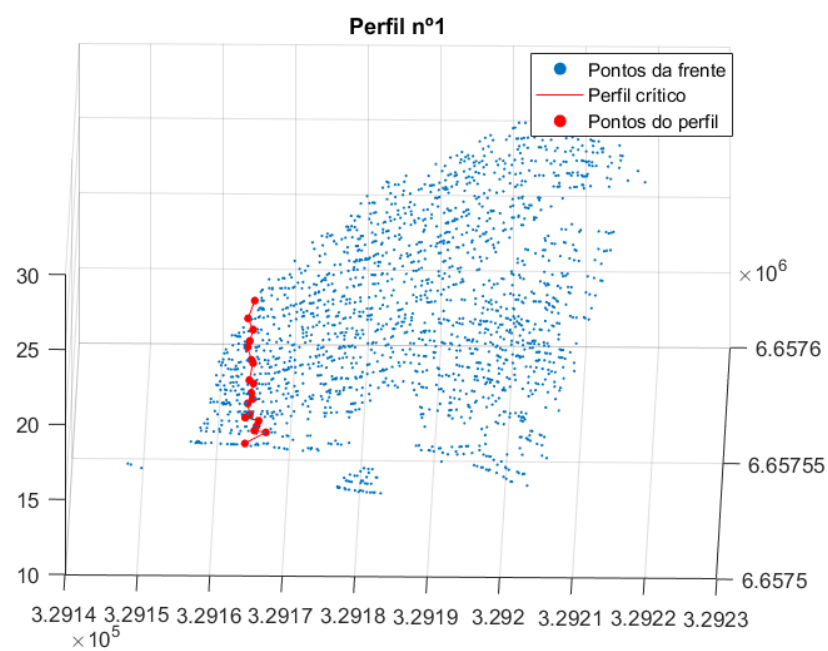


Figura B.2.3. – Perfil nº1 (3D)

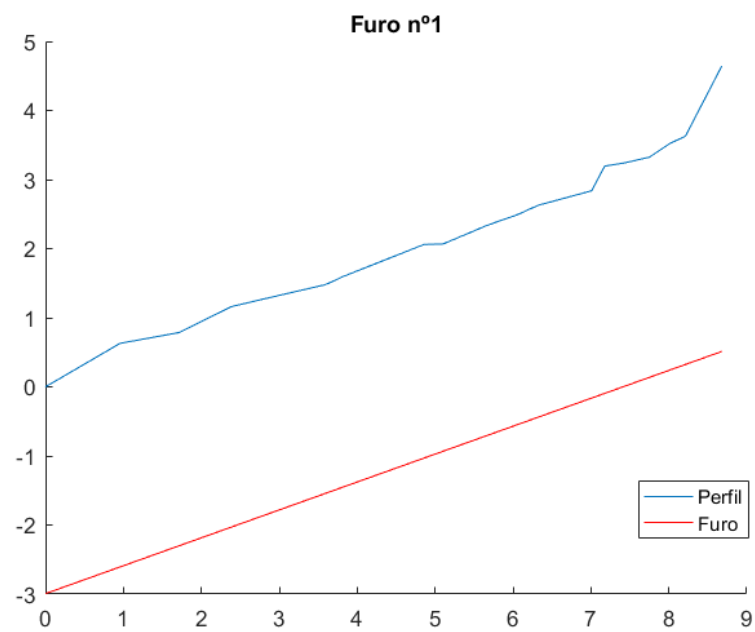


Figura B.2.4. – Furo nº1 (2D)

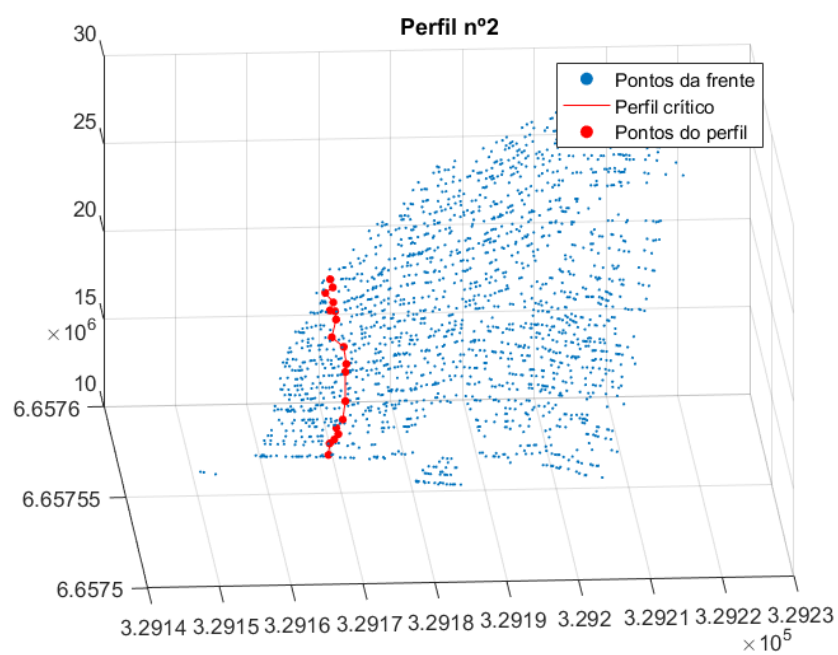


Figura B.2.5. – Perfil nº2 (3D)

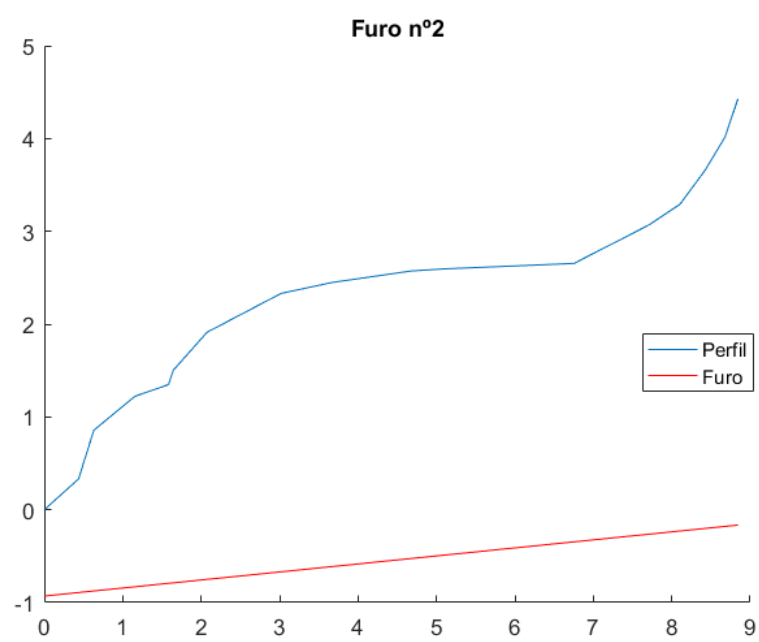


Figura B.2.6. – Furo nº2 (2D)

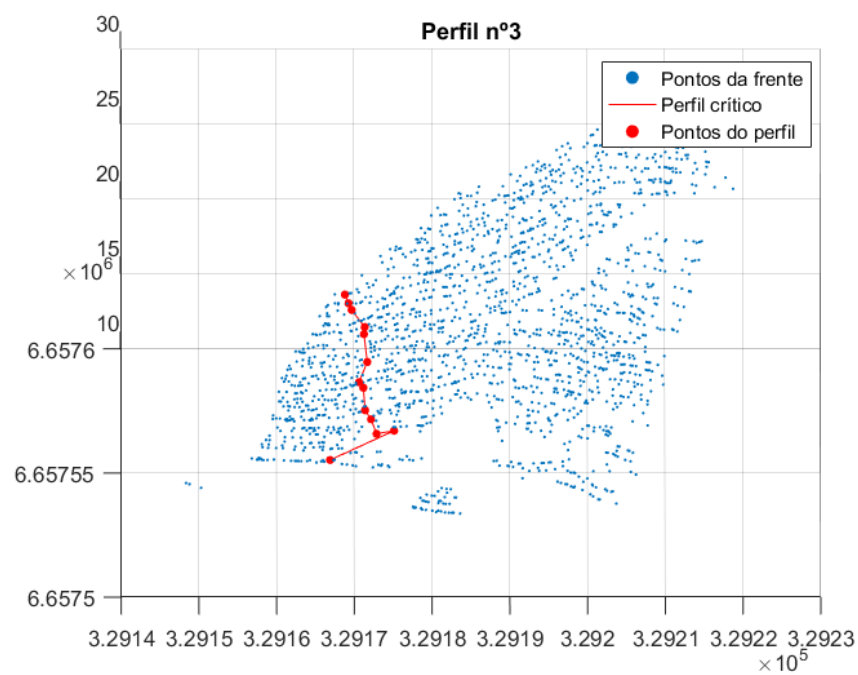


Figura B.2.7. – Perfil nº3 (3D)

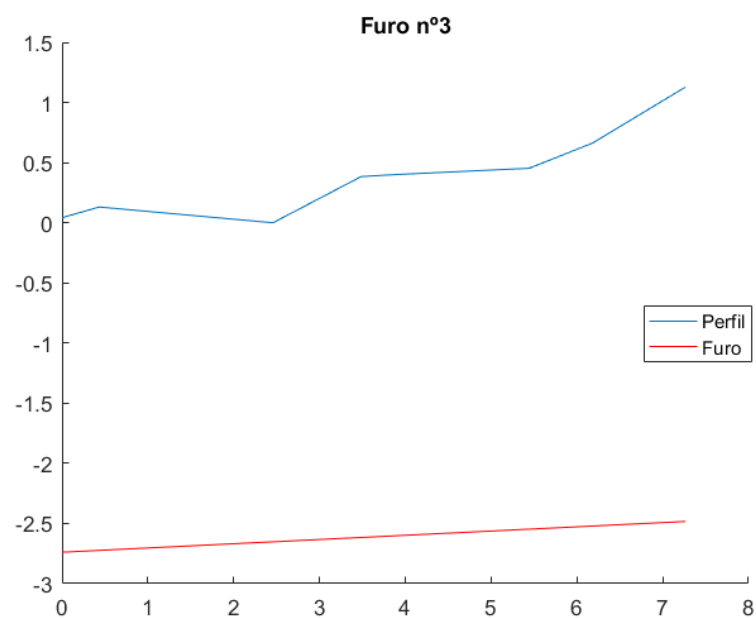


Figura B.2.8. – Furo nº3 (2D)

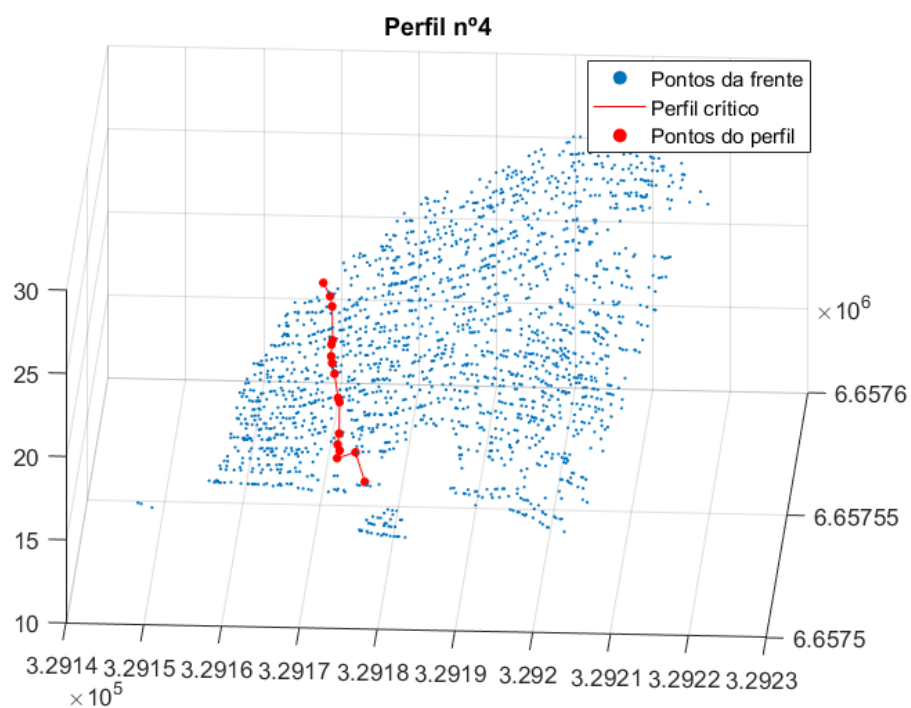


Figura B.2.9. – Perfil nº4 (3D)

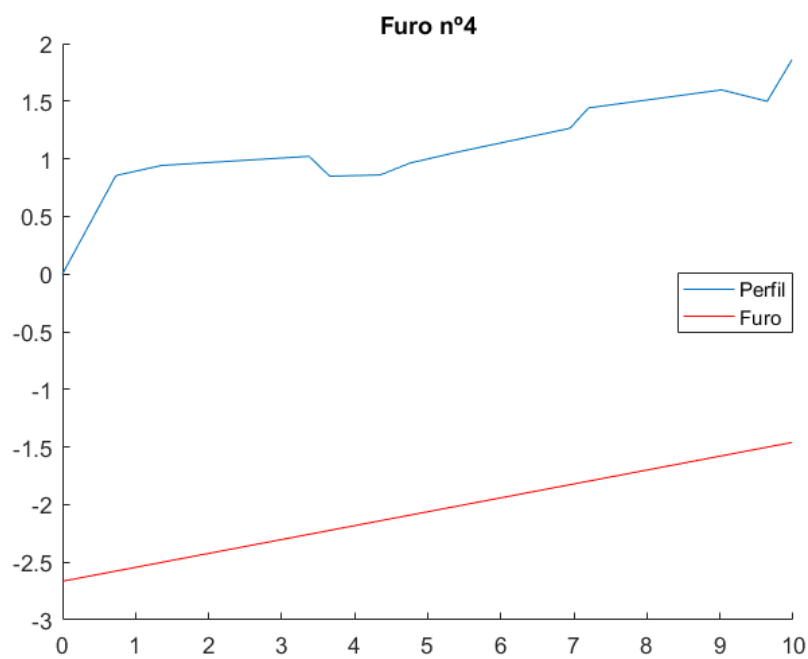


Figura B.2.10. – Furo nº4 (2D)

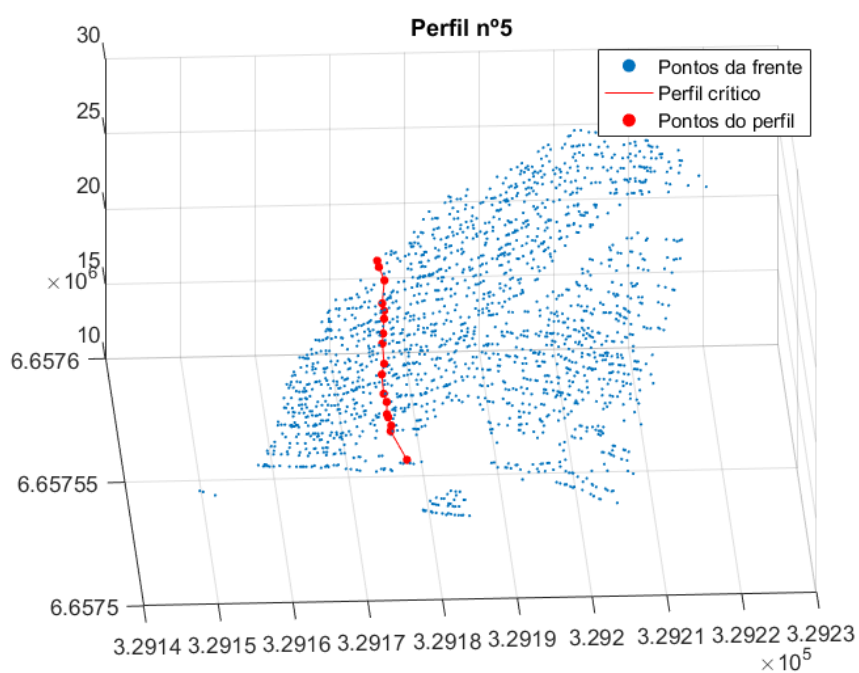


Figura B.2.11. – Perfil nº5 (3D)

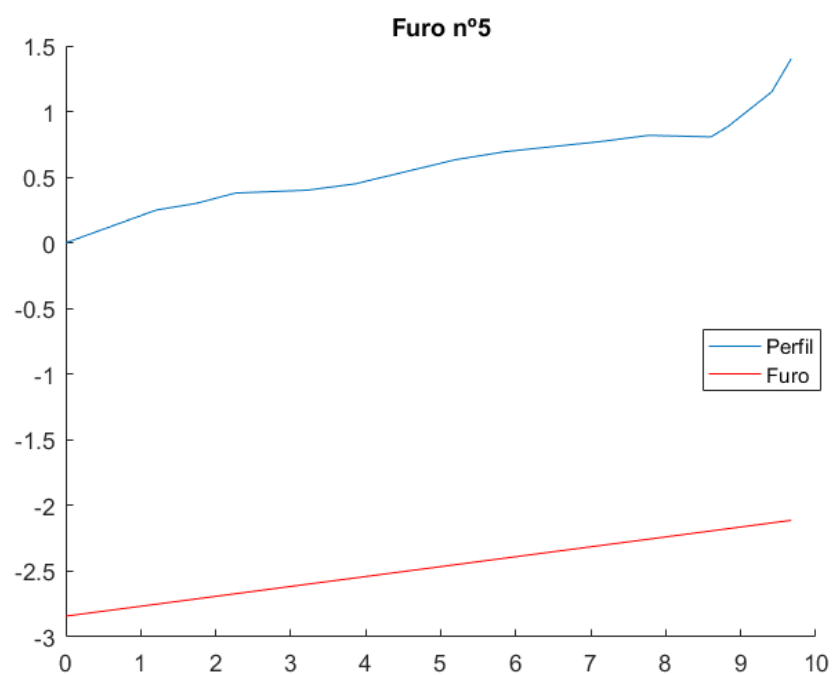


Figura B.2.12. – Furo nº5 (2D)

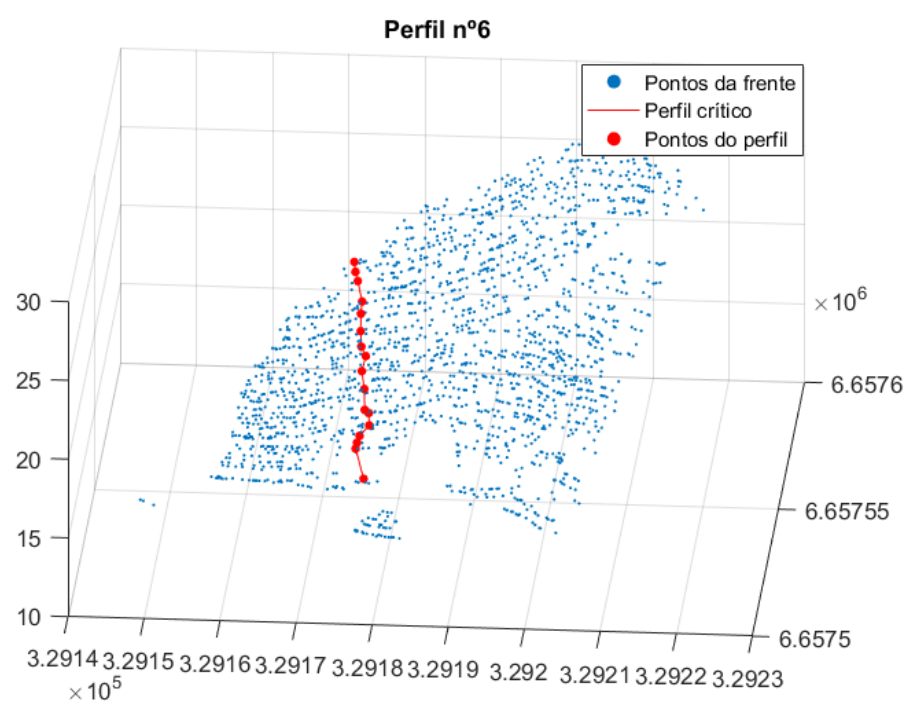


Figura B.2.13. – Perfil nº6 (3D)

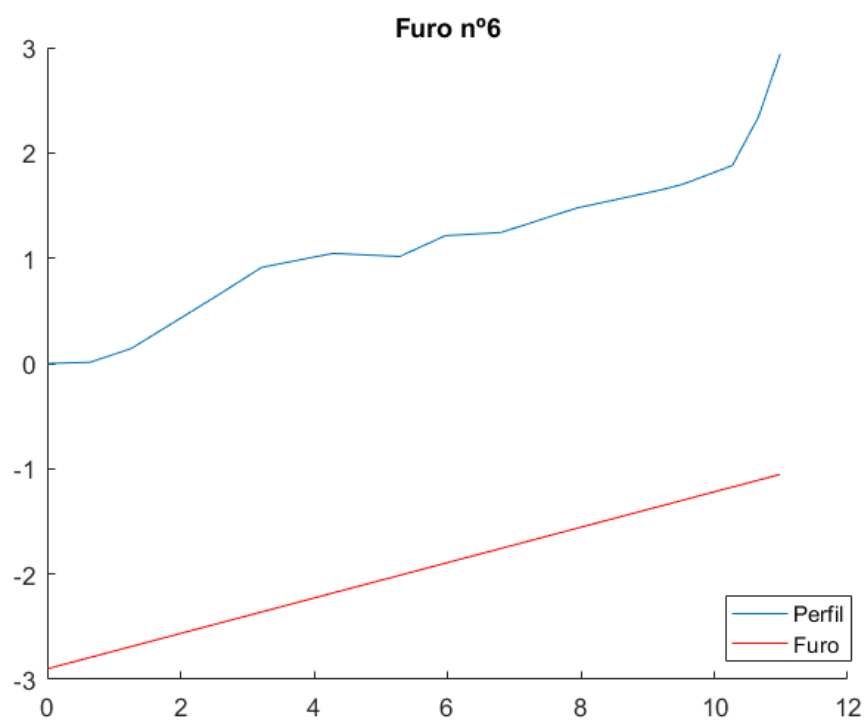


Figura B.2.14. – Furo nº6 (2D)

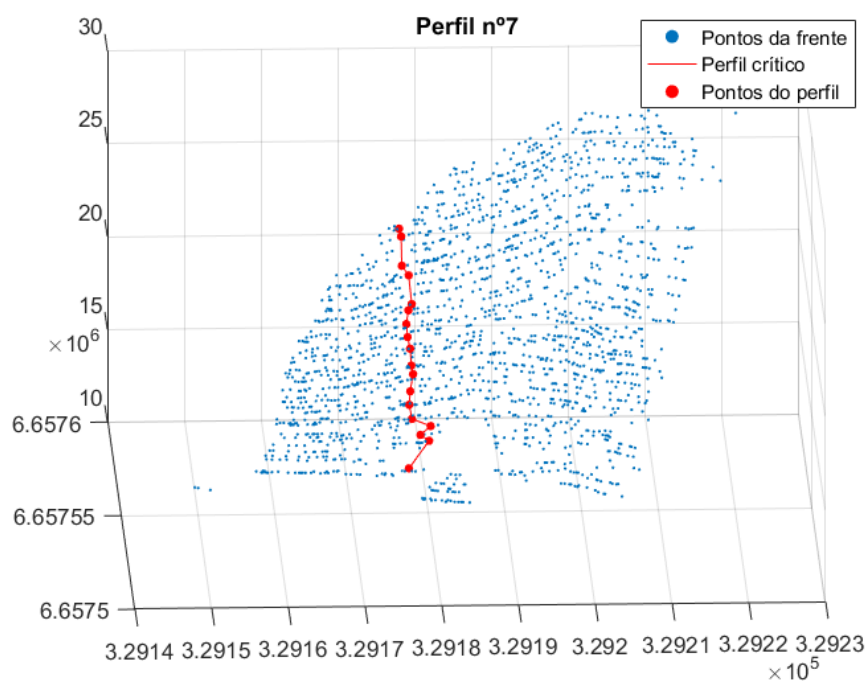


Figura B.2.15. – Perfil nº7 (3D)

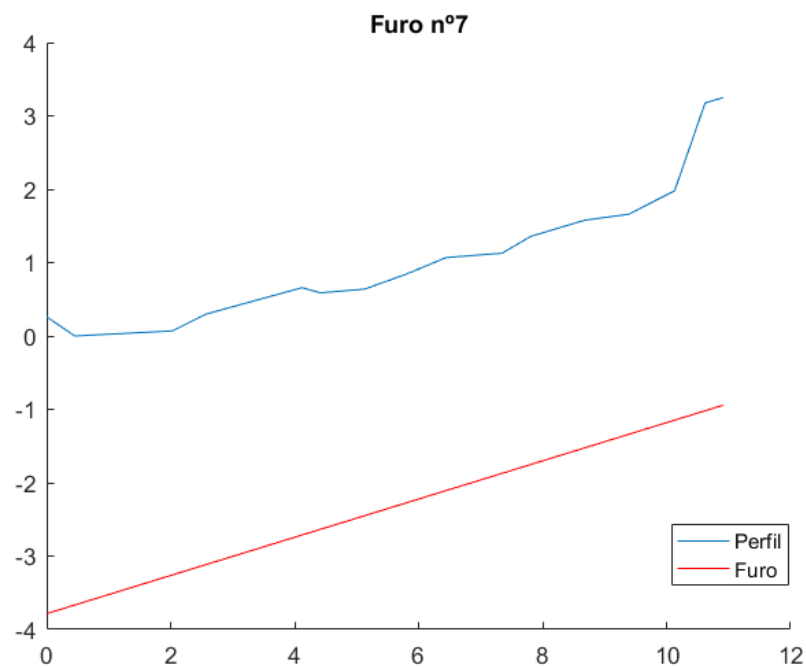


Figura B.2.16. – Furo nº7 (2D)

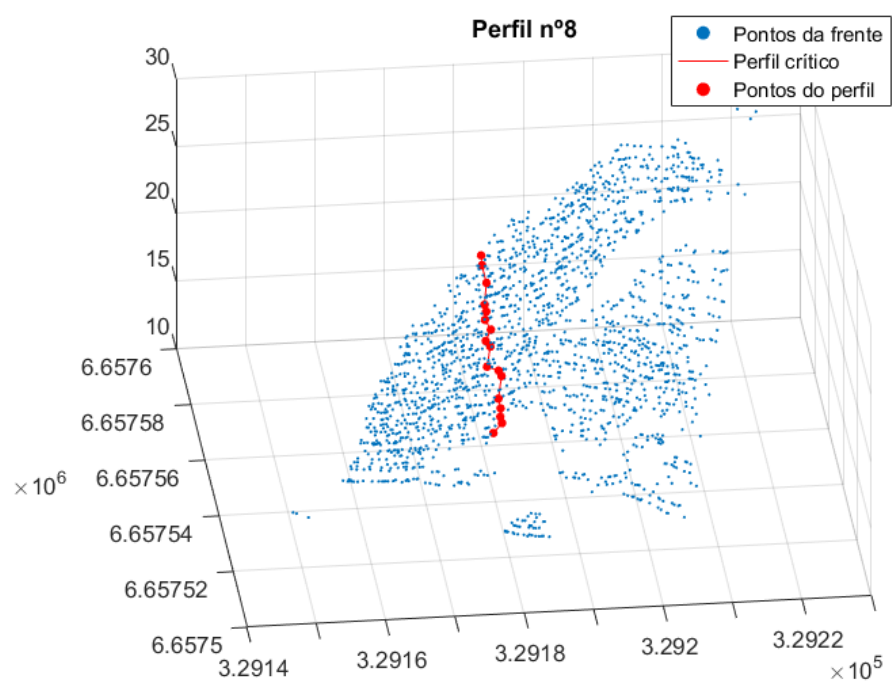


Figura B.2.17. – Perfil nº8 (3D)

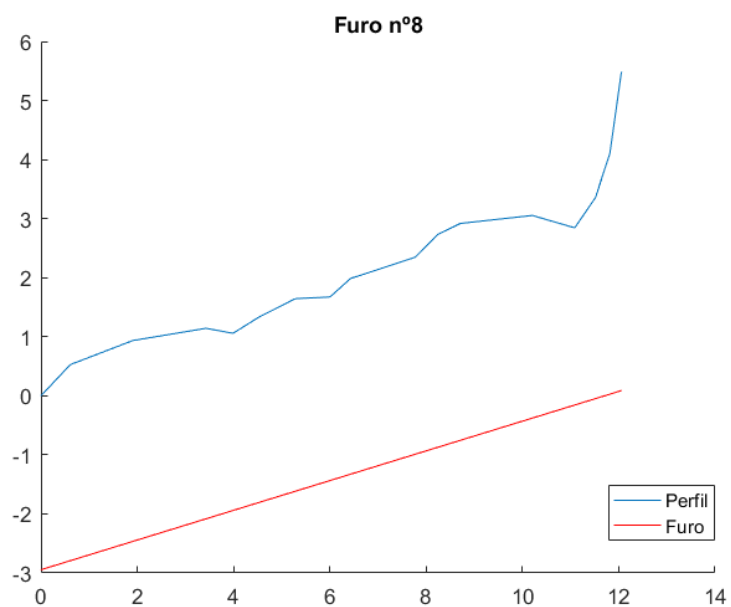


Figura B.2.18. – Furo nº8 (2D)

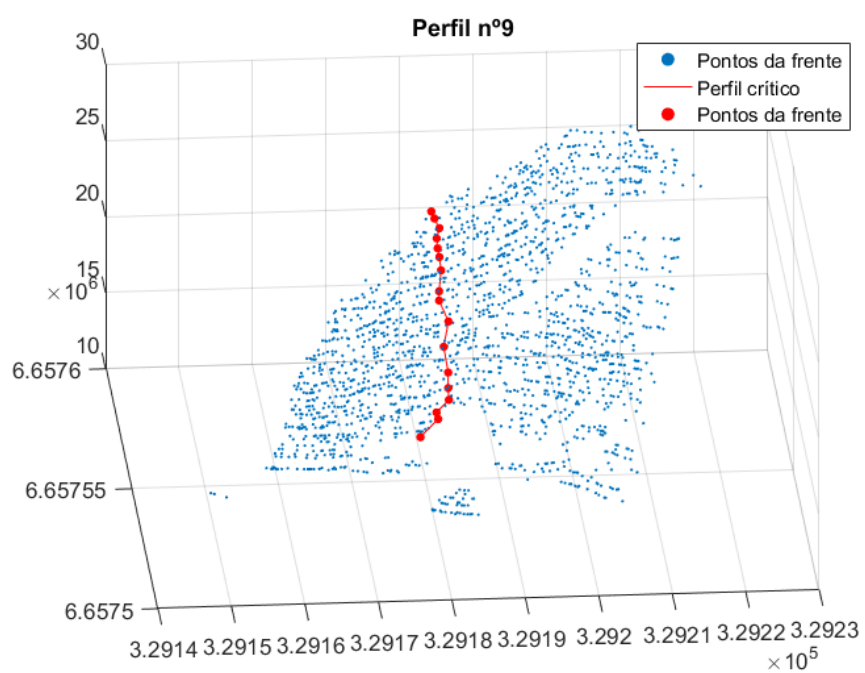


Figura B.2.19. – Perfil nº9 (3D)

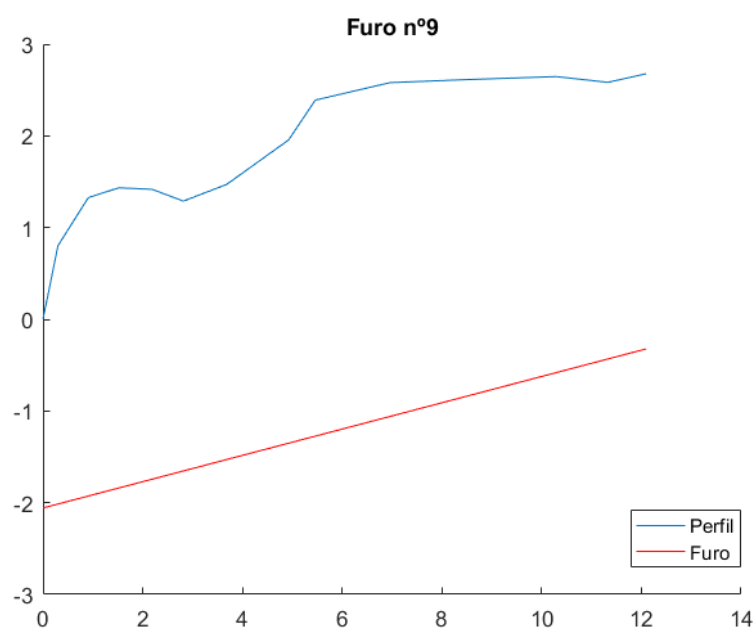


Figura B.2.20. – Furo nº9 (2D)

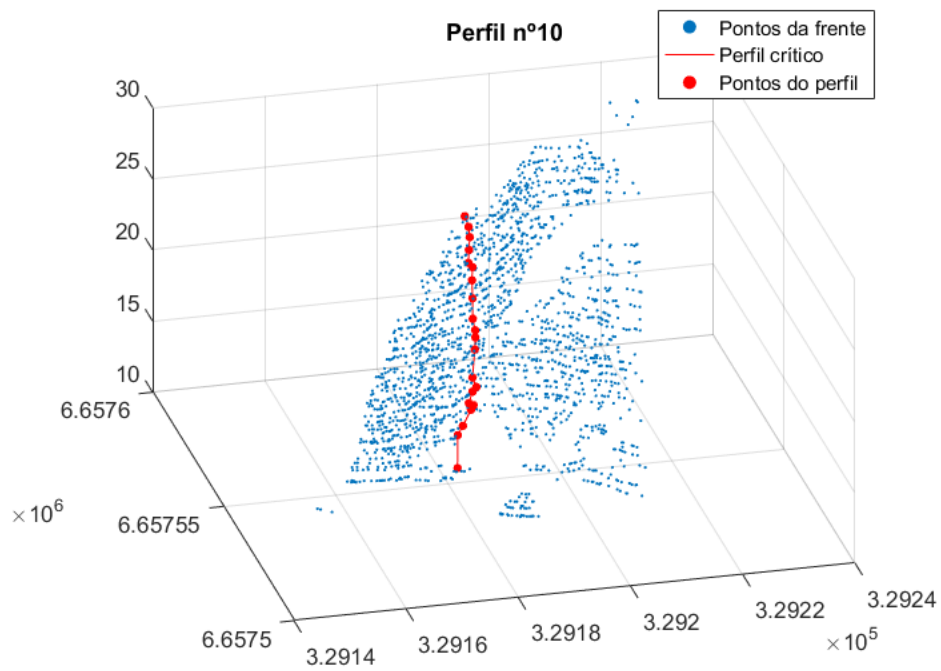


Figura B.2.21. – Perfil nº10 (3D)

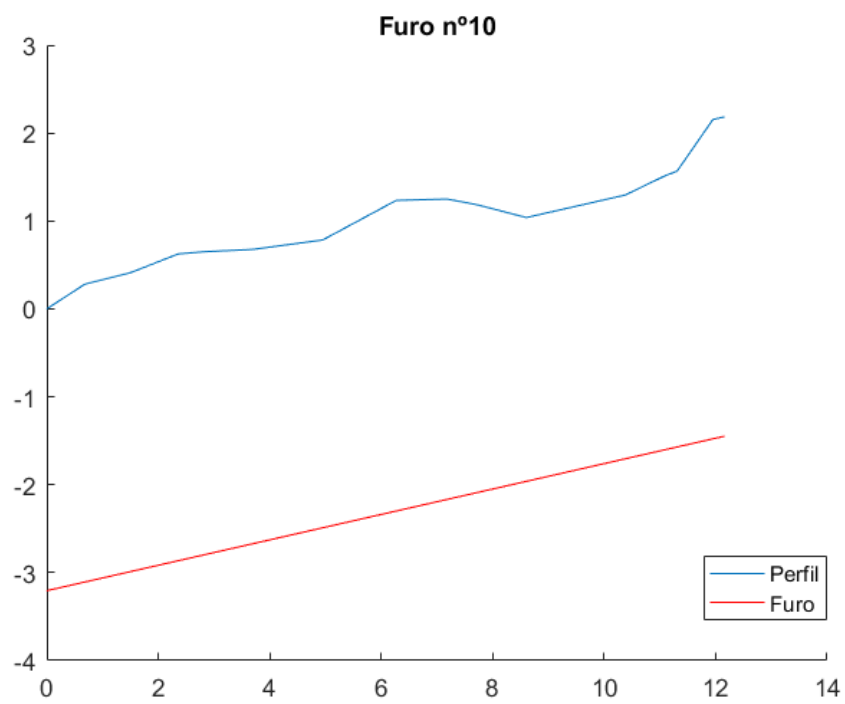


Figura B.2.22. – Furo nº1 (2D)

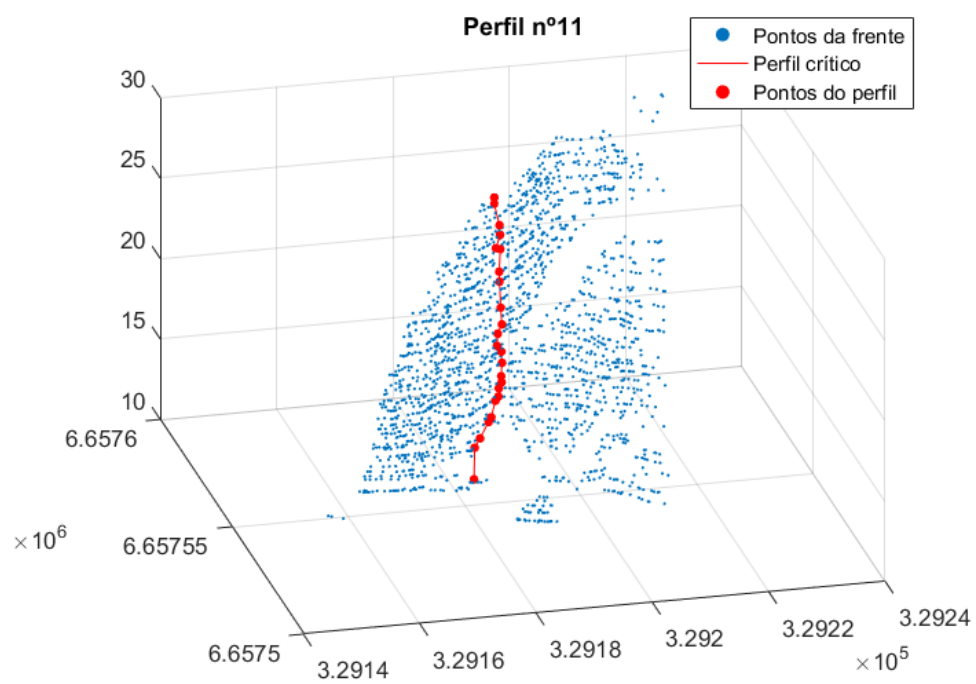


Figura B.2.23. – Perfil nº11 (3D)

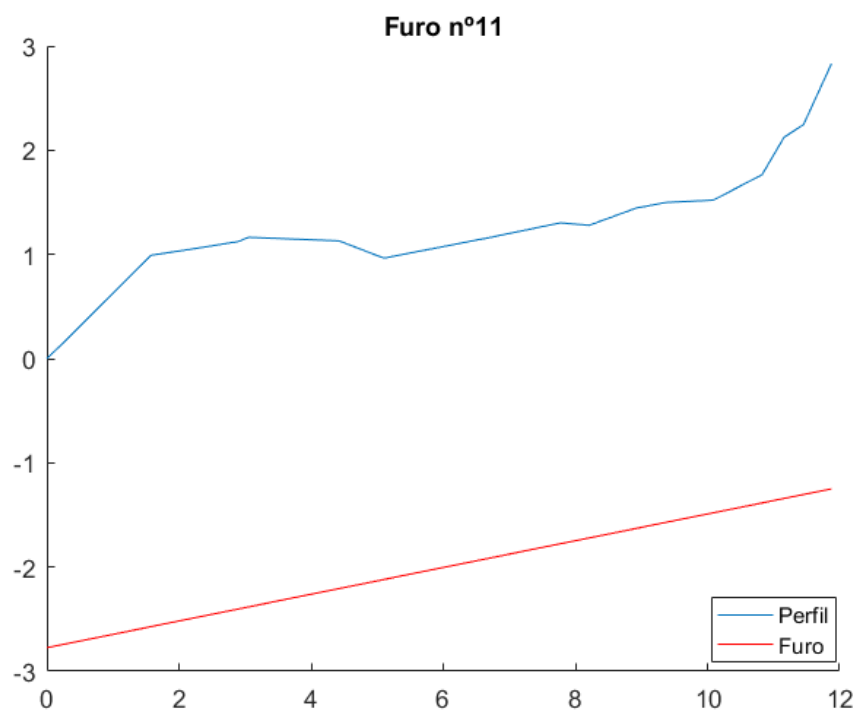


Figura B.2.24. – Furo nº11 (2D)

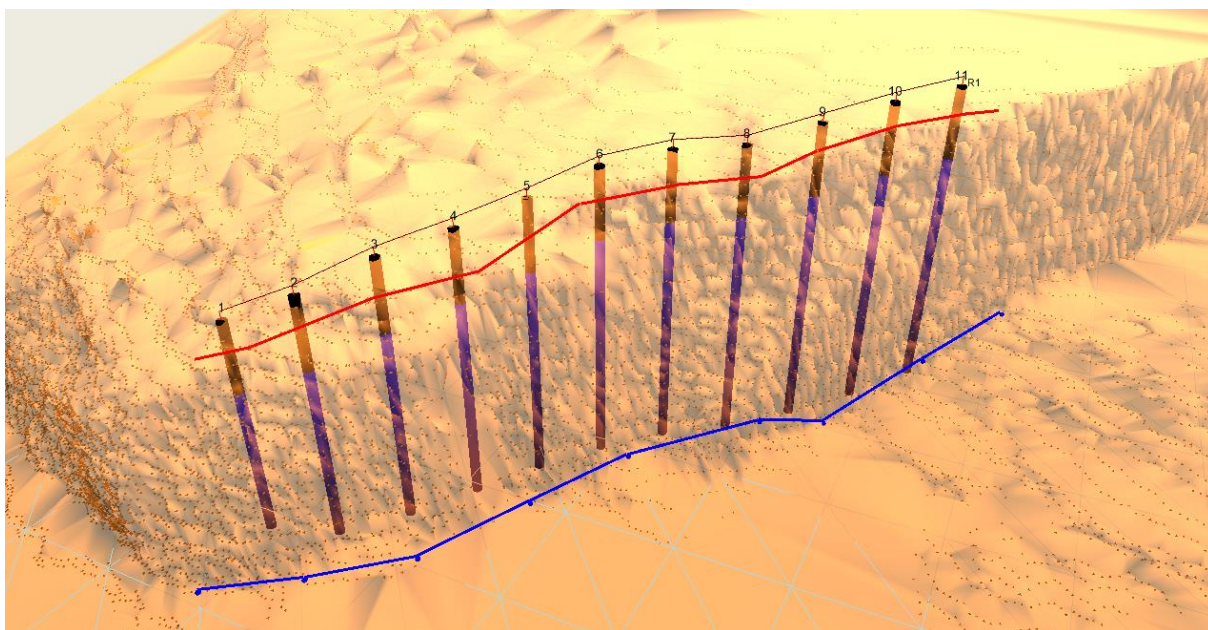


Figura B.2.25 – Bancada com furos-padrão

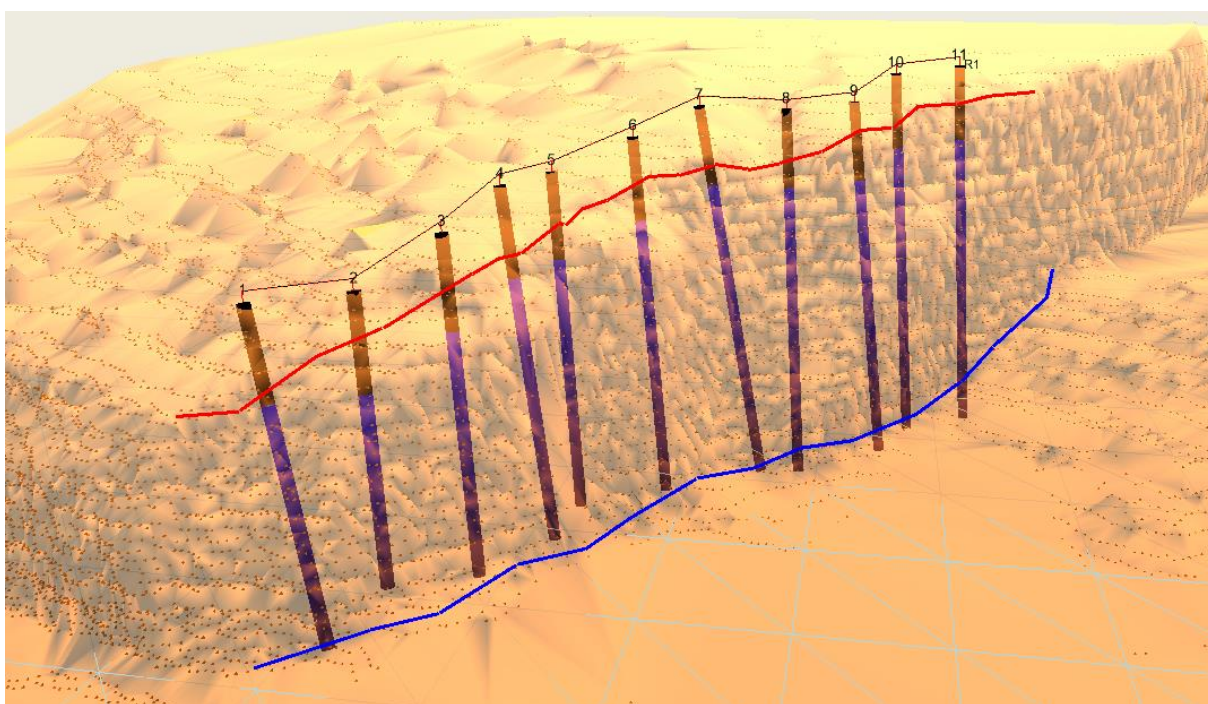


Figura B.2.26 – Bancada com furos otimizados

Uma vez mais, é notória a diferença entre as inclinações dos furos após a otimização. À semelhança do exemplo anterior, é necessário fazer alguns ajustes noutros parâmetros geométricos como o espaçamento e o comprimento dos furos.

